

Konturdetektion bei Herzbildern mit Hilfe des Tensor Voting Verfahrens



Axel Gerstmair

Diplomarbeit
an der Universität Ulm,
Fakultät für Informatik, Abteilung Neuroinformatik

und dem Universitätsklinikum Ulm,
Abteilung Innere II, Kardiologie



Gutachter:
Prof. Dr. Heiko Neumann
Dr. Hans Kestler

November 2003

*„Nicht weil die Dinge schwierig sind, wagen wir sie nicht,
sondern weil wir sie nicht wagen, sind sie schwierig.“*
– Seneca, epistulae morales 104, 26

Die vorliegende Diplomarbeit gab mir die Gelegenheit, ein Anwendungsgebiet der Informatik in einer „fachfremden“ Disziplin, der Medizin, ausführlich kennen zu lernen. Obwohl ich ursprünglich den Schwerpunkt auf der praktischen Anwendung bzw. einem funktionierendem Prototypen sah, musste ich mich zunehmend auf die theoretischen Aspekte des eingesetzten Tensor Voting Verfahrens konzentrieren, um am Ende zu einer qualifizierten Aussage kommen zu können und nicht den Umfang des Themas zu überschreiten. Bei den vielen Ansatzpunkten für weitere Untersuchungen und andere Verfahren wäre dies ein Leichtes gewesen! Dennoch hoffe ich, eine Brücke zwischen Theorie und Praxis geschlagen zu haben, und denke, zu einem interessanten Ergebnis gekommen zu sein.

Ich habe mir Mühe gegeben, neben den notwendigen Fachbegriffen soweit wie möglich einfache Begriffe zu wählen und auf Fremdwörter zu verzichten, um den oftmals anspruchsvollen Text besser lesbar zu machen. Da die Arbeit auf Deutsch verfasst ist, die meisten Fachbegriffe aber aus der englischen Literatur stammen, musste ich diese übersetzen, um eine Anhäufung von Zitaten zu vermeiden. Manche Begriffe mögen anfangs etwas ungewohnt sein, dennoch bin ich der Meinung, dass dies zur Qualität des Textes beiträgt.

Zu Beginn der Diplomarbeit wollte ich ein Glossar zum Nachschlagen aller wichtigen Begriffe erstellen, habe dann jedoch sehr schnell festgestellt, dass ich die meisten Definitionen doppelt aufnehmen musste: einmal im Text und einmal im Glossar. Deshalb ließ ich diese Idee wieder fallen und machte mir die Mühe, einen ausführlichen Index zu erstellen, der auf die Definition (kursiv gedruckt) und Verwendung von Begriffen hinweist.

Das in Anhang A vorgestellte Verfahren der Active Shape Models entdeckte ich während meiner Recherche nach einer geeigneten Methode, die durch das Tensor Voting Verfahren extrahierten Kurvensegmente zusammenzufassen. Ich habe mich einige Zeit mit dem Verfahren beschäftigt und es auf die Herzbilder angewendet, um die Ergebnisse eines komplett anderen Ansatzes zu sehen, obwohl es nicht direkt zum Thema dieser Arbeit gehört. Ich habe den Ansatz mit in den Anhang aufgenommen, weil ich es für zu schade hielt, ihn nicht zu erwähnen.

Danksagung

Ganz besonders bedanken möchte ich mich bei meiner Familie für ihre Unterstützung in jeder Hinsicht, insbesondere jedoch für die Hilfe bei der Fertigstellung der Diplomarbeit – das Ausdrü-

cken und Korrekturlesen und die konstruktive Kritik. Und natürlich bei allen anderen, die mir moralische Unterstützung während des halben Jahres gegeben haben.

Martin Babós danke ich für die „Starthilfe“ und geduldige Hilfestellung bei der Einarbeitung in das Tensor Voting Verfahren, die etwa den ersten Monat der Arbeit in Anspruch genommen hat. Außerdem möchte ich mich bei Sylvain Fischer für den Erfahrungsaustausch während seines Gastaufenthalts in der Abteilung Neuroinformatik und für einige neue Ideen bedanken (siehe dazu auch Kapitel 6).

Ich danke Prof. Dr. Vinzenz Hombach und der gesamten Abteilung der Kardiologie für die Bereitstellung meines Arbeitsplatzes und die freundliche Atmosphäre, sowie für die Bereitstellung der Datensätze der Herzbilder, insbesondere Nico Merkle und Dr. Thorsten Nusser, die stets sehr hilfsbereit waren und mich bei Fragen zum medizinischen Hintergrund unterstützt und mir zahlreiche Tipps gegeben haben.

Mein Dank geht auch an Patrick J. Lynch und Eduard Sojka für die großzügige Bereitstellung ihrer Zeichnungen und Bilder (Abbildungen 3.2, 5.1 und 5.2).

Und nicht zuletzt möchte ich mich besonders bei Prof. Dr. Heiko Neumann und Dr. Hans Kestler für die Betreuung meiner Diplomarbeit bedanken und für die Zeit, die sie in den begleitenden Gesprächen aufgebracht haben.

Ich hoffe, die Arbeit ist angenehm und interessant zu lesen, und wünsche dabei nun viel Kurzweil!

Axel Gerstmair, im November 2003

Vorwort	iii
Inhaltsverzeichnis	v
Abbildungsverzeichnis	ix
1 Einleitung	1
1.1 Aufgabenstellung	1
1.2 Tensor Voting und andere Verfahren	1
1.2.1 Hough-Transformation	2
1.2.2 Regularisierung	3
1.2.3 Konsistentes Labeling	3
1.2.4 Active Shape Models	3
1.3 Grundlagen der Merkmalsextraktion	4
1.3.1 Stufen der Merkmalsextraktion	4
1.3.2 Strukturen in zweidimensionalen Bildern	4
1.4 Lösungsansatz	5
1.5 Gliederung	6
1.6 Notation	6
2 Das Tensor Voting Verfahren	7
2.1 Übersicht	7
2.2 Ein Anwendungsbeispiel	9
2.3 Repräsentation geometrischer Strukturen	10
2.3.1 Die Darstellung als Tensor	10
2.3.2 Die Tensorkodierung	12
2.4 Die Tensorvotierung	12
2.4.1 Ausrichten der Votierfelder	13

Inhaltsverzeichnis

2.4.2	Votierung	14
2.4.3	Zerlegung jedes Tensors in sein Eigensystem	14
2.4.4	Erstellen der Salienzkarten	15
2.5	Votierfelder	15
2.5.1	Das grundlegende, zweidimensionale Votierfeld	16
2.5.2	Das zweidimensionale Ball-Votierfeld	19
2.5.3	Dreidimensionale Votierfelder	20
2.5.4	Der Skalenfaktor	20
2.6	Extraktion geometrischer Strukturen	21
2.6.1	Die Non-maximum Suppression Methode	21
2.6.2	Schwellwertverfahren	22
2.7	Zeitkomplexität	22
3	Tensor Voting auf Grauwertbildern	23
3.1	Ansatz	23
3.2	Einfache Merkmalsextraktion	24
3.2.1	Kantendetektion	24
3.2.2	Eckendetektion	24
3.2.3	Probleme	26
3.3	Polymorphe Merkmalsextraktion	26
3.3.1	Der Strukturtensor	26
3.3.2	Gabor-Filter	28
3.3.3	Probleme	29
3.4	Ergebnisse	29
4	Experimente	31
4.1	Versuchsaufbau	31
4.1.1	Rahmenbedingungen	31
4.1.2	Einflussfaktoren	32
4.1.3	Das Fehlermaß	32
4.2	Das gewöhnliche Tensor Voting Verfahren	33
4.2.1	Der Skalenfaktor der Votierfelder	33
4.2.2	Rauschen	35
4.2.3	Diskretisierungseffekte bei Rotation von Strukturen	37
4.2.4	Überbrückung von Konkavitäten	37
4.3	Tensor Voting gegenüber anderen Verfahren	39
4.3.1	Rampenförmige Kanten	40
4.3.2	Schwache Kontraste	41
4.3.3	Rauschen	42
4.3.4	Bewertung	44

4.4	Zusammenfassung	44
5	Konturdetektion bei Herzbildern	45
5.1	Anatomie des Herzens	45
5.2	Besondere Problematik	46
5.3	Die Konturextraktion im Überblick	48
5.4	Vorverarbeitung	48
5.5	Extraktion von Kurvensegmenten	50
5.5.1	Die Varianzmaske	50
5.6	Auswahl der Kurvensegmente	51
5.6.1	Intensitätsmaske	52
5.6.2	Endokontur des linken Ventrikels	53
5.6.3	Epikontur des linken Ventrikels	53
5.6.4	Konturen des rechten Ventrikels	54
5.7	Gruppierung der Kurvensegmente	54
5.7.1	Gradient Vector Flow	55
5.8	Ergebnisse	55
6	Erweitertes Tensor Voting	59
6.1	Votierung über Phasen des Herzzyklus hinweg	59
6.2	Krümmung	61
6.2.1	Verfahren nach Tang und Medioni	61
6.2.2	Verfahren nach Fischer	64
6.3	Polarität	65
6.4	Mehrere Iterationen des Votierdurchgangs	65
7	Ergebnisse und Ausblick	67
7.1	Zusammenfassung	67
7.2	Mögliche Verbesserungen	68
7.3	Diskussion	69
7.4	Ausblick	70
7.4.1	Multi-scale Tensor Voting	70
7.4.2	Dreidimensionale Rekonstruktion des Herzens	70
A	Active Shape Models	71
A.1	Erzeugung des statistischen Modells	71
A.1.1	Auswahl von Landmarken	71
A.1.2	Ausrichten der Formen	72
A.1.3	Modellierung der Variation	72
A.2	Anpassung der Modellform	73
A.2.1	Ermitteln der neuen Punkte	75

Inhaltsverzeichnis

A.3 Ergebnisse	76
B Hinweise zur Implementierung	79
B.1 Bilineare Interpolation	79
B.2 Parameter des Extraktionsverfahrens	80
B.3 Externe Matlab-Funktionen	80
C Die beiliegende CD	81
Literaturverzeichnis	83
Index	87

Abbildungsverzeichnis

1.1	Ein Originalbild mit manuell ermittelten Konturen.	2
1.2	Die Hough-Transformation.	3
1.3	Unterschiedliche Arten von Punkten.	5
1.4	Unterschiedliche Arten von Kanten.	5
2.1	Übersicht über das Tensor Voting Verfahren.	8
2.2	Die Verarbeitungsschritte des Tensor Voting Verfahrens an einem Beispiel.	9
2.3	Definition der Krümmung.	10
2.4	Anschauliche Darstellung eines 3D-Tensors.	11
2.5	Überlappung zweier Stab-Votierfelder.	13
2.6	Das grundlegende, zweidimensionale Votierfeld.	16
2.7	Ermitteln der Richtung für jeden Punkt im Raum.	17
2.8	Energiefluss zwischen zwei Punkten.	17
2.9	Der 90°-Bereich des Votierfeldes.	18
2.10	Das zweidimensionale Ball-Votierfeld.	19
2.11	Unterschiedliche Perspektiven auf zwei zu überbrückende Kantenelemente.	20
2.12	Das Kanisza-Dreieck.	21
3.1	Vergleich verschiedener Kantendetektoren.	24
3.2	Vergleich verschiedener Eckendetektoren.	25
3.3	Gradientenrichtungen zur Berechnung des Strukturtenors.	27
3.4	Anwendung eines geraden und eines ungeraden Gabor-Filters.	28
3.5	Salienzkarten für Struktur- und Gabor-Tensoren.	30
4.1	Größenvergleich unterschiedlicher Testfiguren und Votierfelder.	33
4.2	Einfluss des Skalenfaktors bezüglich Größe einer Struktur.	34
4.3	Einfluss des Skalenfaktors bezüglich Distanz zweier Strukturen.	35
4.4	Einfluss des Skalenfaktors bezüglich Überlagerung zweier Strukturen.	35

Abbildungsverzeichnis

4.5	Ein Beispiel für „Shot Noise“.	36
4.6	Einfluss von „Shot Noise“ auf das Tensor Voting Verfahren.	36
4.7	Diskretisierungseffekte bei Rotation zweier orthogonaler Linien.	37
4.8	Konstruktion der Testfigur mit Konkavitäten.	38
4.9	Ergebnisse für die Überbrückung von Konkavitäten.	39
4.10	Ergebnisse unterschiedlicher Verfahren bei rampenförmigen Kanten.	41
4.11	Ergebnisse unterschiedlicher Verfahren bei schwachen Kontrasten.	42
4.12	Einfluss von Rauschen auf unterschiedliche Verfahren.	43
5.1	Die Anatomie des Herzens (Ansicht seitlich und von vorne).	45
5.2	Die Anatomie des Herzens (Ansicht der Kurzachse und schematisch).	46
5.3	Problematische Bereiche bei der Konturfindung.	47
5.4	Übersicht über das Verfahren zur Extraktion der Herzkonturen.	49
5.5	Varianzbild und Varianzmaske.	50
5.6	Originalbild und Intensitätsmaske.	52
5.7	Auswahl der inneren Konturen.	53
5.8	Auswahl der äußeren Kontur des linken Ventrikels.	54
5.9	Gradient Vector Flow für die Endokontur des linken Ventrikels.	55
5.10	Beispiele für die Konturextraktion.	57
5.11	Beispiele für die Konturextraktion (Fortsetzung).	58
6.1	Votierung über Phasen des Herzzyklus hinweg.	60
6.2	Tensorvotierung unter Berücksichtigung der Krümmung.	61
6.3	Votieren für die Krümmung.	62
6.4	Spezielle Votierfelder bei Berücksichtigung des Vorzeichens der Krümmung.	64
6.5	Spezielles Votierfeld bei Berücksichtigung der Krümmung nach Fischer.	65
6.6	Veranschaulichung der Polarität.	65
6.7	Mehrere Iterationen der Tensorvotierung.	66
A.1	Änderung der Modellform-Parameter.	73
A.2	Ermitteln eines Intensitätsprofils und Suche nach einer neuen Position.	74
A.3	Initialisierung mit der Referenzform.	75
A.4	Ergebnisse für das Active Shape Model.	76
B.1	Bilineare Interpolation.	79
C.1	Die Verzeichnisstruktur der beiliegenden CD.	81

Erkrankungen des Herzens und seiner Gefäße spielen in der westlichen Gesellschaft eine große Rolle und gehören zu den häufigsten Todesursachen. Zu deren Diagnose kann die Magnetresonanztomographie (MRT)¹ eingesetzt werden, die hochauflösende, dreidimensionale Bilder und Bildfolgen liefert. Die Bilder erlauben die Bestimmung verschiedener, klinisch bedeutsamer Parameter, wie z. B. das Volumen der linken Herzkammer, die Muskelmasse, Wandstärken oder gar die Durchblutung der Herzmuskulatur. Bevor allerdings diese Werte berechnet werden können, müssen in den Bildern die betreffenden Bereiche abgegrenzt und markiert werden – in der Regel von Hand von einer qualifizierten Fachkraft – was eine sehr aufwändige Arbeit darstellt. Eine automatisierte Umsetzung dieser Bildsegmentierung ist deshalb für eine breitere klinische Anwendung nahezu unumgänglich. Diese Diplomarbeit widmet sich genau diesem Thema und der Konturfindung der Herzkammern im Speziellen (siehe Abbildung 1.1).

1.1 Aufgabenstellung

Gegeben ist eine Serie von Schichtaufnahmen des Herzens mittels der Magnetresonanztomographie. Die Bilder sind zweifach angeschnittenen, d. h. um zwei Achsen im Raum gekippt, und werden über einen kompletten Herzzyklus (Diastole → Systole → Diastole) aufgenommen. Auf diese Weise erhält man einen Datensatz von mehreren hundert Bildern.

Das Ziel ist nun eine möglichst selbständige Erkennung der inneren und äußeren Umrisslinien der linken (und rechten) Herzkammer in den Bildern, d. h. eine Erkennung der sogenannten Endo- bzw. Epikonturen. Eine selbständige Erkennung bedeutet, dass eventuell Parameter von Hand eingestellt werden müssen, aber dass möglichst kein Nachbearbeiten der Konturen nötig ist. Die erkannten Konturen sollen über die Schichten und über die Zeit hinweg eine gewisse Ähnlichkeit haben, damit sie später sinnvoll ausgewertet werden können. Grundlage für die Erkennung der Umrisslinien soll das sogenannte Tensor Voting Verfahren sein.

1.2 Tensor Voting und andere Verfahren

Das Tensor Voting Verfahren – oder im Deutschen auch die Tensorvotierung – ist noch relativ jung und wurde in den letzten Jahren von Professor Medioni und seinem Umfeld entwickelt [MLT00].

¹Auch Kernspinresonanztomographie, Nuclear Magnetic Resonance Tomography (NMR) und Magnetic Resonance Imaging (MRI) genannt.

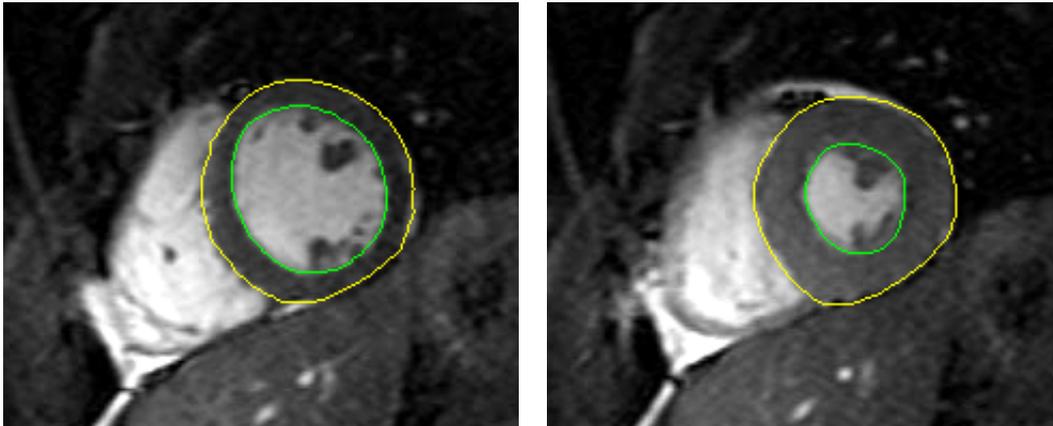


Abbildung 1.1: Ein Originalbild mit manuell ermittelten Konturen. Die beiden Bilder sind Aufnahmen vom Anfang und von der Mitte eines Herzzyklus. Dargestellt sind jeweils die innere und die äußere Kontur der linken Herzkammer.

Es berücksichtigt Grundlagen der Wahrnehmungsgruppierung („perceptual grouping“) aus der Psychologie, um spärliche Eingabedaten zu vervollständigen und um geometrische Merkmale wie Oberflächen, Kurven und Kreuzungspunkte zu extrahieren. Das Verfahren kann als Teil eines Bottom-Up-Ansatzes gesehen werden. Das bedeutet, Eingabebilder werden – wie in Abschnitt 1.3.1 näher beschrieben – stufenweise verarbeitet und in jedem Schritt die gewonnenen Bildmerkmale abstrahiert, bis man am Ende die gesuchte Beschreibung des Bildes erhält. Die Tensorvotierung ist bei diesem Ansatz auf mittlerer Ebene angesiedelt: auf der einen Seite arbeitet sie nicht direkt auf den Rohdaten, sondern bekommt eine vorverarbeitete Eingabe und auf der anderen Seite ist sie ein allgemeines Verfahren, das kein *Modellwissen* berücksichtigt und unabhängig von der Eingabe und dem gesuchten Ergebnis arbeitet. Als *Modellwissen* wird sämtliches Wissen über die Anwendung und die Problemstellung bezeichnet, in dieser Arbeit beispielsweise Wissen über Lage, Form und Beschaffenheit anatomischer Strukturen des Herzens.

Neben der Tensorvotierung sind auch andere Verfahren zur Lösung der hier gegebenen Aufgabenstellung denkbar. Deshalb werden im folgenden vier bekannte Verfahren als andere Wahlmöglichkeiten vorgestellt.

1.2.1 Hough-Transformation

Die Hough-Transformation wurde ursprünglich als Methode zur Erkennung komplexer Punktmuster in binären Bildern eingeführt [IK88]. Dies wird erreicht, indem man Werte für die Parameter bestimmt, die für das Muster charakteristisch sind, z. B. bei einer Geraden den Abstand p vom Ursprung und den Winkel θ der Normalen (siehe Abbildung 1.2). Entscheidend für das Verfahren ist, dass räumlich ausgedehnte Muster im Originalbild kompakte Muster in einem Parameterraum erzeugen. So wird das Problem der globalen Mustererkennung in ein lokales Extremwert-Problem umgewandelt. Veranschaulicht an dem Beispiel der Geraden bedeutet dies, dass die beiden Parameter aller möglichen Geraden durch einen Bildpunkt eine sinusförmige Kurve im Parameterraum erzeugen. Liegen mehrere Punkte im Bild auf einer Linie, so überschneiden sich deren Kurven im Parameterraum und man kann am Schnittpunkt einen bestimmten Abstand p und Winkel θ ablesen und daraus die gesuchte Linie rekonstruieren. Die Hough-Transformation kann verallgemeinert werden, so dass sie Kreise, Ellipsen oder beliebige Formen (auch zusammengesetzte) erkennen kann.

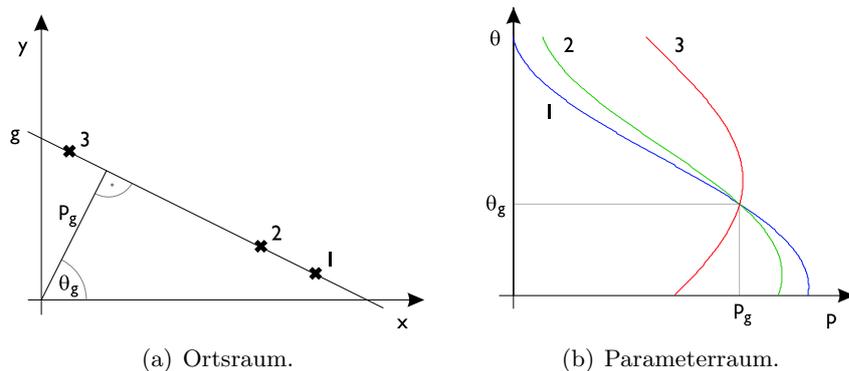


Abbildung 1.2: Die Hough-Transformation. Die Parameter aller möglichen Geraden durch einen Punkt im Ortsraum (a) erzeugen eine sinusförmige Kurve im Parameterraum (Hough-Raum) (b). Am Schnittpunkt der Kurven lassen sich die Parameter der Verbindungsgeraden ablesen.

1.2.2 Regularisierung

Bei der sogenannten Regularisierung [Neu02b] werden Probleme der unteren Bildverarbeitungsstufen als schlecht gestellte („ill-posed“), inverse Probleme aufgefasst. Ein inverses Problem versucht, die Fragestellung zu lösen, bei gegebenen Funktionswerten die zugehörigen Werte aus dem Urbild zu finden. Aufgrund der schlechten Konditionierung des Problems gibt es dafür keine eindeutige Lösung oder die Lösung ist nicht stetig von den gegebenen Daten abhängig (nach Jacques Hadamard, 1902). Um den Suchraum einzuschränken, muss deshalb zusätzliches Wissen herangezogen werden. Meistens werden die Probleme durch Minimierung eines Funktionals gelöst.² Beispiele für Regularisierungsverfahren sind die sogenannten *Snakes* nach Kass u. a. [KWT88] und die darauf basierenden Verfahren. Bei der Konturextraktion bei Herzbildern wird in der zweiten Verarbeitungsstufe ein Regularisierungsverfahren verwendet, der sogenannte Gradient Vector Flow Ansatz (siehe Abschnitt 5.7).

1.2.3 Konsistentes Labeling

Das konsistente Labeling [Neu02b] stammt aus dem Bereich der künstlichen Intelligenz und wird dort meistens als Constraint Satisfaction Problem [RN95] bezeichnet. Es geht darum, alle Bildpunkte widerspruchsfrei mit Kennzeichnungen zu versehen und so beispielsweise Konturen zu markieren. Gelöst wird das Problem durch einen iterativen Prozess, der eine Initialisierung, eine „Update“-Regel und eine Abbruchbedingung erfordert. Die Hauptprobleme dabei sind zum einen der riesige Suchraum und zum anderen die Schwierigkeit, eine widerspruchsfreie Lösung zu finden.

1.2.4 Active Shape Models

Beim Verfahren der Active Shape Models (ASM) nach Cootes u. a. [CT01] handelt es sich um ein statistisches Verfahren, das eine Modellform anhand von Trainingsdaten erstellt. Durch die Ermittlung charakteristischer Verformungen, wird beim Aufbau des Modells die Anzahl der freien Parameter drastisch verringert. Die Modellform wird anschließend in der eigentlichen Anwendung schrittweise an die Gegebenheiten des Bildes angepasst. Diese Vorgehensweise stellt einen Top-Down-Ansatz dar und steht damit dem Bottom-Up-Ansatz der Tensorvotierung gegenüber. In Anhang A ist ein kleiner Exkurs zu finden, der etwas ausführlicher auf die Active Shape Models

²Ein Funktional ist eine reellwertige Funktion, die auf einem Vektorraum \mathcal{V} definiert ist: $f : \mathcal{V} \rightarrow \mathbb{R}$. \mathcal{V} kann insbesondere auch ein Funktionenraum sein, d. h. man kann sich f als eine Funktion vorstellen, die auf Funktionen definiert ist.

eingeht. Es gibt weitere, ähnliche Verfahren, wie z. B. das Active Appearance Model (AAM), ebenfalls in [CT01] beschrieben, das zusätzlich zur Form von Strukturen deren Textureigenschaften berücksichtigt, oder das Verfahren nach Wang und Staib [WS00], das das statistische Modell mit einem Modell glatter Kurven kombiniert.

1.3 Grundlagen der Merkmalsextraktion

Dieser Abschnitt erläutert einige Grundlagen der Merkmalsextraktion und führt Begriffe ein, die in der weiteren Ausarbeitung benutzt werden.

1.3.1 Stufen der Merkmalsextraktion

Entsprechend des klassischen, dreistufigen Bildverarbeitungssystems unterscheidet man die folgenden Arten der Merkmalsextraktion und die zugehörigen Verarbeitungsprozesse [FH98]:

- *Low-level* Prozesse (z. B. Bildrestauration) arbeiten auf dem diskreten Bildbereich, d. h. Ein- und Ausgabe sind normale Bilder. *Ikonische Merkmale* sind Eigenschaften von Pixeln, wie z. B. Farbe, Gradient oder Textur.
- *Mid-level* Prozesse dienen dazu, von der bildhaften Darstellung zu einer symbolischen zu abstrahieren. Dabei werden *geometrische Merkmale* (Punkte, Linien, Regionen) gewonnen, die im Bild besonders herausragende Strukturen darstellen. Man kann anschließend geometrische Merkmale zu höherwertigen *Strukturmerkmalen* zusammenfassen, beispielsweise mehrere Linien zu einem Polygonzug, jedoch noch ohne semantisches Wissen (Wissen über das Bild) zu benutzen.
- *High-level* Prozesse nutzen Messmethoden und semantisches Wissen, um einzelnen Merkmalen eine Bedeutung zuweisen zu können – man erhält *semantische Merkmale*, z. B. die Region der linken Herzkammer in einem Herzquerschnittsbild.

1.3.2 Strukturen in zweidimensionalen Bildern

Bei der Merkmalsextraktion nutzt man die Intensitätswerte in einem Bild, um bestimmte geometrische Merkmale (Punkte, Kanten und Regionen) erkennen und unterscheiden zu können. In zweidimensionalen Bildern lassen sich diese Merkmale in drei Kategorien einteilen: die intrinsisch zwei-, ein- und nulldimensionalen Strukturen. Je nach Kategorie gibt es zwei, eine oder keine Vorzugsrichtungen, die das Merkmal festlegen. Innerhalb dieser Kategorien kann man noch feinere Einteilungen treffen (siehe Fuchs und Heuel [FH98]), was für die symbolische und semantische Weiterverarbeitung oftmals nötig ist, um die gewünschten Ergebnisse zu erhalten.

Intrinsisch zweidimensionale Strukturen

Punkte, *Endpunkte* von Linien, *Ecken* und *Kreuzungspunkte* sind intrinsisch zweidimensionale Strukturen. Alle diese Merkmale können durch ein Koordinatenpaar (x, y) vollständig bestimmt werden, jedoch kann man anhand der Umgebung unterschiedliche Bedeutungen zuordnen (siehe Abbildung 1.3).

Endpunkte (Anfangs- und Endpunkte von Linien), Ecken (Schnittpunkte genau zweier Linien) und Kreuzungspunkte (Schnittpunkte von drei oder mehr Linien) werden zur Beschreibung von Kanten und Rändern von Regionen benutzt. Fehlen einige dieser Merkmale, so kann dies schwerwiegende Folgen für die symbolische Beschreibung des Bildes haben.

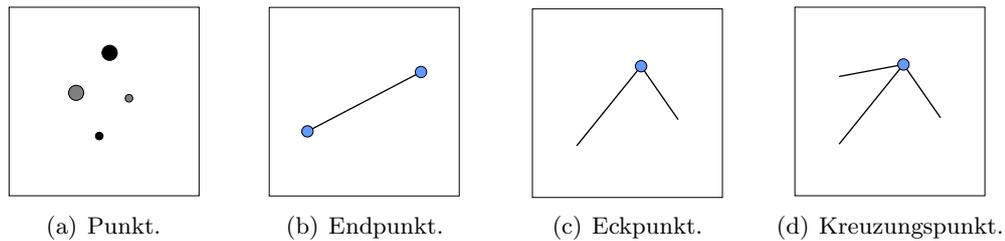


Abbildung 1.3: Unterschiedliche Arten von Punkten.

Intrinsisch eindimensionale Strukturen

Kanten und *Linien* sind intrinsisch eindimensionale Strukturen – sie können durch ein Koordinatenpaar und eine Vorzugsrichtung (die Tangentenrichtung) festgelegt werden. Will man den Unterschied treffen, dann sind Kanten die Grenzlinien zweier Regionen und Linien schmale, lange Objekte, z. B. ein dunkler Balken auf hellem Untergrund oder umgekehrt.

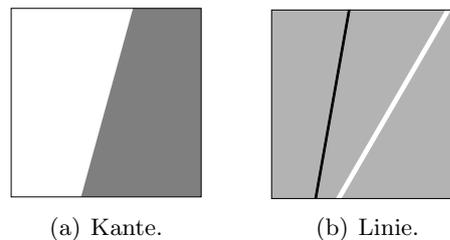


Abbildung 1.4: Unterschiedliche Arten von Kanten.

Intrinsisch nulldimensionale Strukturen

Man bezeichnet *Regionen* als intrinsisch nulldimensionale Strukturen, da sie keine Vorzugsrichtung haben. Regionen sind Bildbereiche, die ein bestimmtes Ähnlichkeitsmaß, ein Homogenitätskriterium, erfüllen, wie z. B. der Helligkeitswert eines Pixels oder die Textureigenschaften seiner Umgebung. Regionen können durch ihre Umrisslinien repräsentiert werden.

1.4 Lösungsansatz

Entsprechend der oben beschriebenen Stufen der Merkmalsextraktion verfolgt diese Arbeit folgenden Lösungsansatz, wobei die mittlere Stufe in zwei Abschnitte unterteilt wurde:

1. *Vorverarbeitung.* Verbesserung der Qualität des gesamten Bildes oder der Regionen, die von Interesse sind, z. B. Anpassung von Helligkeit und Kontrast oder Beseitigung von Rauschen durch Anwendung eines Tiefpassfilters. Dieser Schritt ist in Abschnitt 5.4 erläutert.
2. *Extraktion geometrischer Merkmale.* Extraktion von Punkten und Kanten, die als Eingabe für das Tensor Voting Verfahren dienen. Hier kommt das in Kapitel 3 beschriebene Strukturtensor-Verfahren zum Einsatz.
3. *Verfeinerung durch Tensor Voting.* Verbesserung der Qualität von extrahierten Strukturen und Zusammenfassen mehrerer geometrischer Merkmale zu größeren Einheiten (Verbindung von Kurvensegmenten). Dies ist Thema der Kapitel 2, 4 und 6.

1 Einleitung

4. *Extraktion semantischer Merkmale.* Glättung der extrahierten Strukturen und Ermittlung der gesuchten Konturen. Damit wird den Kurven auch eine Bedeutung zugeordnet, z. B. Endokontur der linken Herzkammer. Diese letzte Verarbeitungsstufe ist in Kapitel 5 beschrieben.

1.5 Gliederung

Die meisten Kapitel dieser Arbeit bauen aufeinander auf und sind entsprechend angeordnet. Kapitel 2 und 3 bilden die Grundlage für alle weiteren Kapitel. Kapitel 2 führt in das Thema der Tensorvotierung ein, das anschließend in Kapitel 3 in der Anwendung auf Grauwertbilder fortgeführt wird. Es folgt Kapitel 4 mit zahlreichen Experimenten und Auswertungen, die einige Eigenschaften des Tensor Voting Verfahrens, wie z. B. den Skalenfaktor der Votierfelder oder den Umgang mit Rauschen, untersuchen. Kapitel 5 stellt schließlich das mehrstufige Verfahren zur Konturextraktion bei Herzbildern vor. Spezielle und allgemeine Erweiterungen für das Tensor Voting Verfahren werden anschließend in Kapitel 6 dargestellt. Das siebte und letzte Kapitel fasst die Ergebnisse der Diplomarbeit zusammen und gibt einen Ausblick auf mögliche Verbesserungen und Erweiterungen.

In Anhang A ist ein Exkurs über Active Shape Models zu finden, der in aller Kürze das Verfahren vorstellt und Ergebnisse aufzeigt. Einige Bemerkungen zur Implementierung der Tensorvotierung sind in Anhang B aufgeführt, der Inhalt der beiliegenden CD in Anhang C.

1.6 Notation

Soweit nicht anders angegeben, werden folgende mathematische Schreibweisen verwendet:

Symbol	Bedeutung
α, β, γ	skalärer Wert
a, b, c	skalärer <i>oder</i> vektorieller Wert
$\hat{a}, \hat{b}, \hat{c}$	Vektor der Einheitslänge
\mathbf{T}	Tensor (zweiter Ordnung)
V^S	Stab-Votierfeld
\tilde{V}^S	Stab-Votierfeld als Vektorfeld mit Hauptrichtung der Tensoren
V^B	Ball-Votierfeld
\tilde{V}^B	Ball-Votierfeld als Vektorfeld mit Hauptrichtung der Tensoren
$g(x, y)$	Zweidimensionale Grauwertfunktion eines Bildes
$g(p)$	Zweidimensionale Grauwertfunktion eines Bildes mit Vektor p
$g_x(\cdot)$	Erste partielle Ableitung der Grauwertfunktion nach x : $\frac{\partial g(\cdot)}{\partial x}$
$g_{xy}(\cdot)$	Zweite partielle Ableitung der Grauwertfunktion nach x und y : $\frac{\partial^2 g(\cdot)}{\partial x \partial y}$
$\nabla g(\cdot)$	Grauwertgradient der Bildfunktion: $(\frac{\partial g(\cdot)}{\partial x} \quad \frac{\partial g(\cdot)}{\partial y})^\top$

Tabelle 1.1: Notation

Das Tensor Voting Verfahren

Dieses Kapitel stellt die Grundlagen des Tensor Voting Verfahrens vor. Die Darstellungen und Erläuterungen basieren auf dem Buch von Medioni, Lee und Tang [MLT00], auf den Doktorarbeiten von Lee [Lee98] und Tang [Tan00], sowie auf einem Artikel von Guy und Medioni [GM96]. Das Kapitel beginnt mit einer kurzen Übersicht über das gesamte Verfahren, in der jedoch noch nicht auf Details eingegangen wird. Die einzelnen Schritte des Verfahrens und Begriffe, die in diesem Zusammenhang gebraucht werden, werden dann in den darauffolgenden Abschnitten ausführlicher beschrieben.

2.1 Übersicht

Das Tensor Voting Verfahren nutzt Aspekte aus der Wahrnehmungspsychologie, um kleine Strukturen in einem Bild zusammenzufassen, und extrahiert dann geometrische Merkmale wie Oberflächen, Kurven und Kreuzungspunkte. Es lässt sich auf beliebig viele Dimensionen verallgemeinern, jedoch beschränkt sich hier die Beschreibung auf den zweidimensionalen (und teilweise dreidimensionalen) Fall. Das Verfahren gliedert sich in mehrere Schritte, die in Abbildung 2.1 dargestellt sind. Es setzt einen Vorverarbeitungsschritt voraus, der aus einem Bild die sogenannten *Eingabemerkmale* extrahiert (in der Abbildung nicht sichtbar, siehe Kapitel 3). Ein Eingabemerkmale kann ein Punkt, ein *Kurvenelement* oder – im dreidimensionalen Raum – ein *Flächenelement* sein, d. h. es sind zumindest seine Koordinaten bekannt. Bei einem Kurvenelement kennt man außerdem die Tangentenrichtung und bei einem Flächenelement die Normalenrichtung. Die Anzahl der Eingabemerkmale kann relativ gering sein, da das Verfahren sehr gut mit spärlichen Eingaben umgehen und diese ergänzen kann.

Nachdem nun alle Eingabemerkmale extrahiert worden sind, müssen sie als *Tensoren* zweiter Ordnung kodiert werden, die mathematisch als Matrix dargestellt werden. Beispielsweise wird ein Punkt in einen richtungsunabhängigen (*isotropen*) Tensor mit Einheitsradius umgewandelt. Durch diesen Schritt erhält man eine spärlich besetzte *Tensorkarte*, die z. B. als ein zweidimensionales Feld repräsentiert wird, in dem jede Zelle einen Tensor (bzw. das Nullelement $\mathbf{0}$) enthält und einem ursprünglichen Bildpunkt entspricht.

Es folgt eine erste Votierphase, in der alle Elemente innerhalb einer bestimmten Nachbarschaft miteinander kommunizieren und so ihren Informationsgehalt verbessern. In dieser Votierphase werden nur die bereits bestehenden Elemente verfeinert, jedoch in den leeren Zellen der Tensorkarte noch keine neuen berechnet. Die Tensoren nehmen eine allgemeine (im zweidimensionalen

2 Das Tensor Voting Verfahren

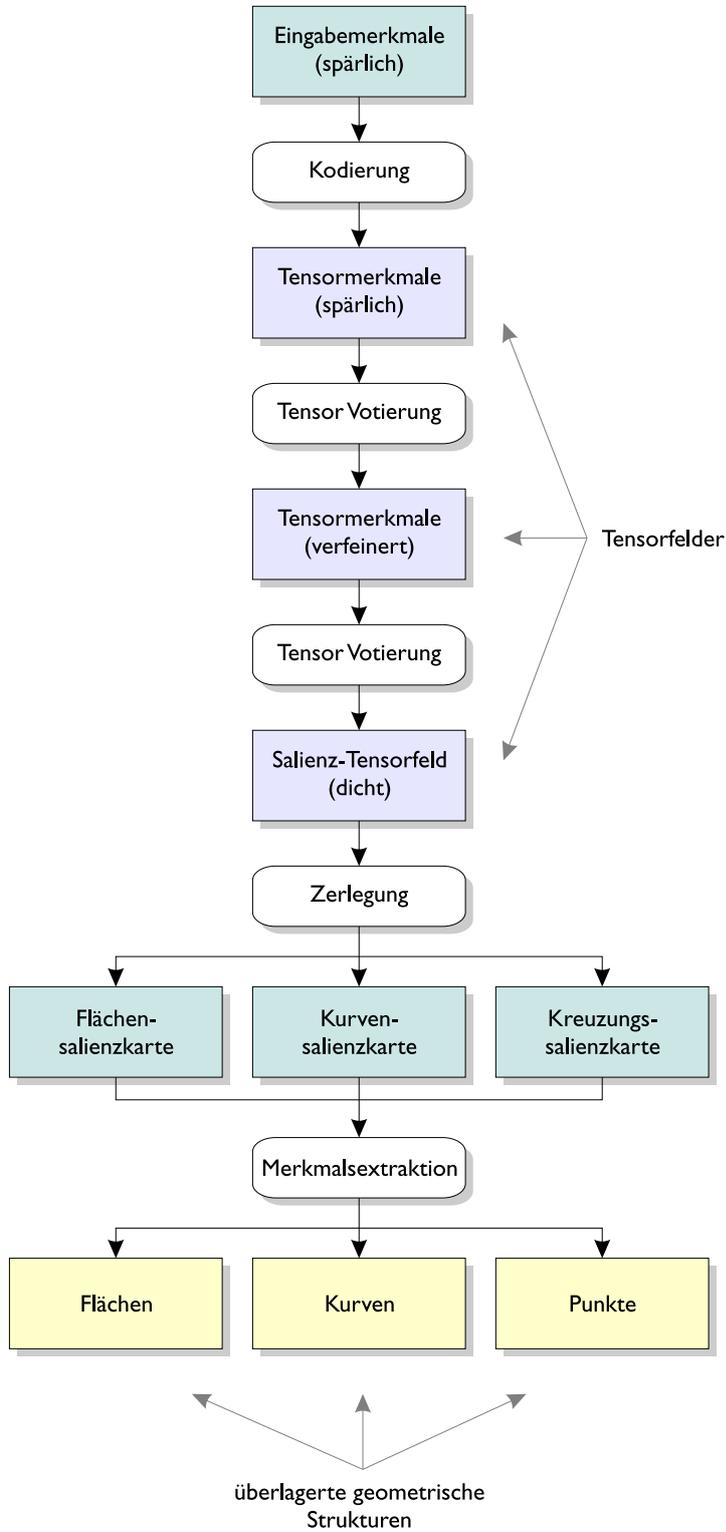


Abbildung 2.1: Übersicht über das Tensor Voting Verfahren.

Fall eine elliptische) Form an, die die zugrunde liegende Struktur beschreibt und die Gewissheit darüber widerspiegelt, z. B. die Ausrichtung eines Kurvenelements.

In der zweiten Votierphase wird die spärlich besetzte Tensorkarte verfeinert und vollständig ausgefüllt, indem nun für *jede* Koordinate bzw. Zelle wiederum innerhalb einer bestimmten Nachbarschaft die geometrischen Merkmale (Punkt-, Kurven- und Flächenartigkeit) ermittelt werden. Die Größe einer Zelle ist dabei variabel und kann je nach gewünschter Genauigkeit gewählt werden (die Genauigkeit ist außerdem abhängig von dem sogenannten Skalenfaktor σ_v , siehe Abschnitt 2.5.4). Der Einfachheit halber kann man als Zelleneinheit das Pixel wählen.

Anschließend müssen die Tensoren in der soeben erhaltenen, dicht besetzten Karte in ihre einzelnen Komponenten zerlegt werden, d. h. in einen Ball- und einen Stabtensor im zweidimensionalen Fall. Die Zerlegung liefert mehrere *Salienzkarten*, die an jeder Koordinate für jedes geometrische Merkmal die Gewissheit angeben („an dieser Stelle befindet sich mit einer Gewissheit von 90 % ein Kurvenelement“).

Im letzten Schritt müssen die einzelnen Salienzkarten mittels Algorithmen der Merkmalsextraktion (siehe Abschnitt 2.6) analysiert werden, wodurch man schließlich die gewünschte Bildbeschreibung erhält.

2.2 Ein Anwendungsbeispiel

An dieser Stelle soll ein Beispiel die Funktionsweise des Tensor Voting Verfahrens verdeutlichen. Als Eingabe dient ein binäres Bild mit Punkten, die bestimmte Figuren andeuten (siehe Abbildung 2.2). Diese Stellen werden als einfache, ungerichtete Balltensoren kodiert, da noch keine Richtungsinformationen vorhanden sind. Die Richtungsinformationen müssen dann in einem ersten Votierdurchgang mit dem sogenannten Ball-Votierfeld grob ermittelt werden. Das entstandene Tensorfeld ist immer noch spärlich besetzt. Nun folgt die zweite Votierphase, in der die neu gewonnenen Richtungsinformationen zur Votierung mit dem sogenannten Stab-Votierfeld genutzt werden. Es entsteht eine dicht besetzte Tensorkarte, aus der man abschließend die Kurven- und Kreuzungssalienen ermitteln und in einer Karte darstellen kann. Die Abbildungen 2.2(b) und (c) zeigen das Ergebnis nach diesem Schritt. Aus den Salienzkarten kann man nun die einzelnen Figuren extrahieren: ein Oval, eine einzelne und zwei sich kreuzende Linien, zu sehen in Abbildung 2.2(d). Die Kreuzungssalienzkarte liefert einen Kreuzungspunkt in der Mitte der sich überschneidenden Linien (nicht dargestellt). Das Beispiel zeigt sehr schön, wie aus nur wenigen, andeutenden Punkten vollständige Figuren extrahiert werden können.

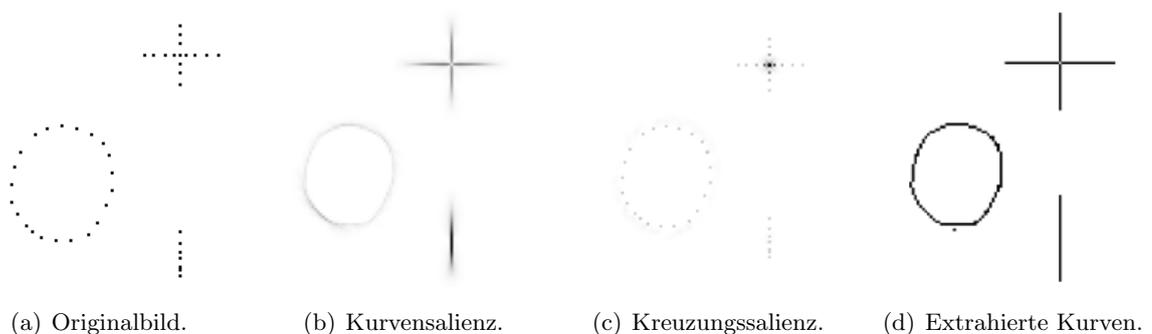


Abbildung 2.2: Die Verarbeitungsschritte des Tensor Voting Verfahrens an einem Beispiel. Die Salienzkarten zeigen potentielle Kurven (b) und Kreuzungspunkte (c). Die Kurven, die aus (b) gewonnen wurden, sind in (d) zu sehen.

2.3 Repräsentation geometrischer Strukturen

Das Ziel des Tensor Voting Verfahrens ist es, geometrische Strukturen in einem Bild zu bestimmen, beispielsweise Regionen, Kurven, Flächen und deren Überschneidungen. Diese sind mathematisch beschrieben und haben unterschiedliche Eigenschaften, die in der Differentialgeometrie untersucht werden [NF90, Bär01]. Punkte werden einfach durch ihre Koordinaten beschrieben. Kurven können durch Punkte und die anliegenden Tangenten dargestellt werden und Flächen durch Punkte und die zugehörigen Normalen.

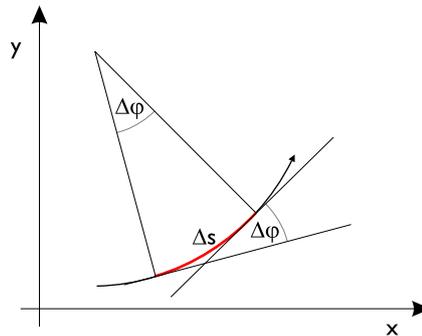


Abbildung 2.3: Definition der Krümmung.

Für Kurven kann man außerdem die sogenannte *Krümmung* bestimmen. Diese ist eine natürliche Invariante ebener Kurven und damit unabhängig vom Koordinatensystem. Bei einem Kreis entspricht die Krümmung dem Kehrwert des Radius r ($\rho = 1/r$). Allgemein ist sie von dem Kontingenzwinkel $d\varphi$ zwischen zwei Tangenten in um ds voneinander entfernten Punkten abhängig [Ste01], wie in Abbildung 2.3 zu sehen ist:

$$\rho(s) = \lim_{\Delta s \rightarrow 0} \frac{\Delta \varphi}{\Delta s} = \frac{d\varphi}{ds} \quad (2.1)$$

Für Flächen gibt es entsprechend zwei sogenannte *Hauptkrümmungen* mit *Hauptkrümmungsrichtungen* [Bär01].

Beim Tensor Voting Verfahren weiß man allerdings nicht im voraus, zu welcher Struktur (Punkt, Kurve oder Fläche) ein Eingabemerkmale gehört. Außerdem können sich geometrische Strukturen überlagern, so dass ein Eingabemerkmale zu mehr als einer Struktur gehört, z. B. zu einem Punkt und einer Kurve, wenn sich zwei Kurven schneiden. Deshalb beschränkt man sich bei der Darstellung nicht auf ein einzelnes geometrisches Merkmal, sondern definiert und ermittelt jeweils ein Maß, das die Gewissheit über das jeweilige Merkmal angibt, die sogenannte *Salienz*.¹

Dieser Ansatz erfordert nun die Darstellung sowohl der geschätzten Parameter der jeweiligen geometrischen Strukturen als auch der damit verbundenen Salienzen für jede Stelle – und zwar möglichst in einer kompakten Form.

2.3.1 Die Darstellung als Tensor

Medioni et al. [MLT00] verwenden in ihrem Verfahren einen symmetrischen *Tensor* zweiter Ordnung. Ein Tensor n -ter Ordnung ist ein mathematisches Objekt mit n Indizes und verallgemeinert damit Skalare (die keinen Index haben), Vektoren (mit einem Index) und Matrizen (mit zwei Indizes) zu Objekten mit einer beliebigen Anzahl an Dimensionen [Lip96]. Die hier verwendeten Tensoren zweiter Ordnung werden also als Matrizen dargestellt und sagen sowohl etwas über ein

¹Etwa: das Herausragen. „Wie sicher bin ich mir darüber, dass es sich hierbei um einen Punkt oder ein Kurvenelement handelt?“

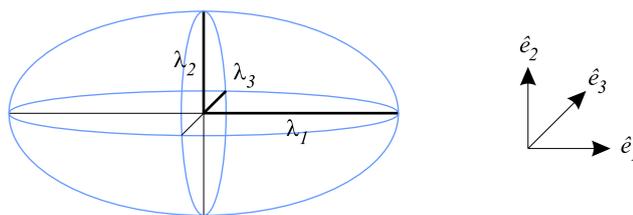


Abbildung 2.4: Anschauliche Darstellung eines 3D-Tensors.

geometrisches Merkmal aus (z. B. es handelt sich um ein Kurvenelement) als auch über dessen Salienz, d. h. über die Gewissheit, die man darüber hat. Sie können im zweidimensionalen Fall als Ellipse, im dreidimensionalen als Ellipsoid dargestellt werden (siehe Abbildung 2.4), wobei die *Form* etwas über die Struktur eines Merkmals aussagt. Je stärker die Darstellung von der Kreis- bzw. Kugelform abweicht, d. h. je exzentrischer diese ist, desto stärker ist der Hinweis auf eine Kurve (bei einer Scheibenform) oder auf eine Fläche (bei einer Stabform). Die *Größe* der Ausdehnung spiegelt dagegen die Salienz wider.

Ein Tensor zweiter Ordnung wird folgendermaßen mathematisch dargestellt:

$$\mathbf{T} = \begin{pmatrix} \hat{e}_1 & \hat{e}_2 & \hat{e}_3 \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \begin{pmatrix} \hat{e}_1^T \\ \hat{e}_2^T \\ \hat{e}_3^T \end{pmatrix} \quad (2.2)$$

$$= \lambda_1 \hat{e}_1 \hat{e}_1^T + \lambda_2 \hat{e}_2 \hat{e}_2^T + \lambda_3 \hat{e}_3 \hat{e}_3^T \quad (2.3)$$

Dabei sind λ_1 , λ_2 und λ_3 die Eigenwerte und \hat{e}_1 , \hat{e}_2 und \hat{e}_3 die entsprechenden normalisierten Eigenvektoren. Die Eigenwerte sind reell und positiv (oder gleich Null), während die Eigenvektoren eine Orthonormalbasis bilden. Die Eigenvektoren zeigen in die Hauptrichtungen des Ellipsoids (bzw. der Ellipse), dessen Größe und Form von den Eigenwerten bestimmt wird. Für einen Tensor mit seinen Parametern schreibt man kurz: $\mathbf{T}(\lambda_1, \lambda_2, \lambda_3, \hat{e}_1, \hat{e}_2, \hat{e}_3)$.

Anstelle der drei Eigenvektoren kann man auch drei Rotationswinkel θ , ϕ und ψ um die z -, y - und x -Achse angeben. Man erhält so eine Rotationsmatrix $R_{\psi\phi\theta}$, die das Weltkoordinatensystem auf das lokale ausrichtet. In diesem Fall beschreibt man den Tensor mit $\mathbf{T}(\lambda_1, \lambda_2, \lambda_3, \theta, \phi, \psi)$.

Komponentendarstellung

Tensoren lassen sich in einzelne Komponenten zerlegen, was den Umgang mit ihnen teilweise erheblich vereinfacht und gewisse Vorteile bringt. So lassen sich beispielsweise aus der umgeformten Darstellung die geometrischen Merkmale unmittelbar herauslesen. Sei nun der Tensor $\mathbf{T}(\lambda_1, \lambda_2, \lambda_3, \hat{e}_1, \hat{e}_2, \hat{e}_3)$ im dreidimensionalen Raum gegeben (entsprechend im zweidimensionalen Raum), dann kann man \mathbf{T} folgendermaßen umschreiben:

$$\mathbf{T} = (\lambda_1 - \lambda_2) \hat{e}_1 \hat{e}_1^T + (\lambda_2 - \lambda_3) (\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T) + \lambda_3 (\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T + \hat{e}_3 \hat{e}_3^T) \quad (2.4)$$

Die einzelnen Komponenten haben folgende Bedeutung:

- $(\lambda_1 - \lambda_2) \hat{e}_1 \hat{e}_1^T$ heißt *Stabkomponente* und entspricht der Flächenartigkeit, wobei \hat{e}_1 die Normalenrichtung angibt.
- $(\lambda_2 - \lambda_3) (\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T)$ heißt *Scheibenkomponente* und entspricht der Kurvenartigkeit, wobei \hat{e}_3 die Tangentenrichtung angibt.
- $\lambda_3 (\hat{e}_1 \hat{e}_1^T + \hat{e}_2 \hat{e}_2^T + \hat{e}_3 \hat{e}_3^T)$ heißt *Ballkomponente* und entspricht der Punktartigkeit.

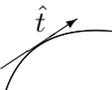
Eingabemerkmale	Belegung der Parameter	Tensor
 Punkt	$\lambda_1 = \lambda_2 = 1, \quad \hat{e}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \hat{e}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$	 Ball
 Kurvenelement	$\lambda_1 = 1, \lambda_2 = 0, \quad \hat{e}_1 = \hat{t} = \begin{pmatrix} t_x \\ t_y \end{pmatrix}, \quad \hat{e}_2 = \begin{pmatrix} -t_y \\ t_x \end{pmatrix}$	 Stab

Tabelle 2.1: Kodierung zweidimensionaler Eingabemerkmale.

2.3.2 Die Tensorkodierung

Dieser Abschnitt gibt eine Übersicht über die Kodierung von Eingabemerkmale als Tensoren. Je nach Art des Eingabemerkmals muss man die Parameter des zugehörigen Tensors wählen, wobei zwischen dem zwei- und dreidimensionalen Raum unterschieden werden muss. Die Tabellen 2.1 und 2.2 fassen die Belegung der Parameter für die einzelnen Eingabemerkmale zusammen.

▷ **Beispiel 2.1** Sei im zweidimensionalen Raum ein Kurvenelement mit der Tangente $\hat{t} = \frac{1}{\sqrt{5}}(2 \ 1)^T$ gegeben, dann muss man die Kodierung anhand der zweiten Zeile in Tabelle 2.1 vornehmen, d. h. die Eigenwerte $\lambda_1 = 1$ und $\lambda_2 = 0$ setzen und die Eigenvektoren mit $\hat{e}_1 = \hat{t} = \frac{1}{\sqrt{5}}(2 \ 1)^T$ und $\hat{e}_2 = \frac{1}{\sqrt{5}}(-1 \ 2)^T$ belegen. Setzt man diese Variablen in Gleichung 2.2 ein, dann ergibt sich folgender Tensor:

$$\mathbf{T} = \frac{1}{\sqrt{5}} \begin{pmatrix} 2 & -1 \\ 1 & 2 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \cdot \frac{1}{\sqrt{5}} \begin{pmatrix} 2 & 1 \\ -1 & 2 \end{pmatrix} = \frac{1}{5} \begin{pmatrix} 4 & 2 \\ 2 & 1 \end{pmatrix}$$

2.4 Die Tensorvotierung

Bei der Tensorvotierung übt jedes Element einen Einfluss auf eine bestimmte Nachbarschaft aus und verfeinert dadurch das Wissen über die geometrischen Merkmale umliegender Elemente. Durch die gegenseitige Beeinflussung verstärken sich geometrische Strukturen im Bild, während willkürliche Erscheinungen, wie z. B. Bildstörungen, abgeschwächt werden. Man kann die Tensorvotierung mit einer Faltung vergleichen, jedoch wird nicht mit skalaren Werten, sondern mit Tensoren gerechnet. Eine wesentliche Rolle bei der Votierung spielen die sogenannten Votierfelder (siehe Abschnitt 2.5), die den Einfluss auf andere Elemente bestimmen. Anstatt für jeden Tensor ein individuelles Votierfeld zu erstellen, genügt es, mit zwei bzw. drei grundlegenden Votierfeldern zu arbeiten, denn man kann den Tensor in einzelne Komponenten zerlegen und mit diesen dann getrennt abstimmen.

Der Votierprozess unterteilt sich in folgende Schritte, die anschließend im Einzelnen beschrieben werden:

1. Ausrichten der Votierfelder
2. Votierung
3. Zerlegung jedes Tensors in sein Eigensystem
4. Erstellen der Salienzkarten

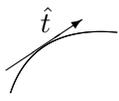
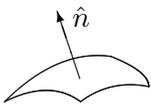
Eingabemerkmale	Belegung der Parameter	Tensor
 Punkt	$\lambda_1 = \lambda_2 = \lambda_3 = 1,$ $\hat{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \hat{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \hat{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$	 Ball
 Kurvenelement	$\lambda_1 = \lambda_2 = 1, \lambda_3 = 0,$ $\hat{e}_1 = \begin{pmatrix} t_z \\ t_y \\ -t_x \end{pmatrix}, \hat{e}_2 = \begin{pmatrix} -t_y \\ t_z \\ -t_x \end{pmatrix}, \hat{e}_3 = \hat{t} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$	 Scheibe
 Flächenelement	$\lambda_1 = 1, \lambda_2 = \lambda_3 = 0,$ $\hat{e}_1 = \hat{n} = \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix}, \hat{e}_2 = \begin{pmatrix} -n_y \\ n_x \\ n_z \end{pmatrix}, \hat{e}_3 = \begin{pmatrix} -n_y \\ -n_z \\ n_x \end{pmatrix}$	 Stab

Tabelle 2.2: Kodierung dreidimensionaler Eingabemerkmale. Bei Kurven- und Flächenelementen ist jeweils ein Eigenvektor durch den Tangenten- bzw. den Normalenvektor vorgegeben. Die beiden anderen stehen senkrecht dazu und können durch eine entsprechende Rotation berechnet werden.

2.4.1 Ausrichten der Votierfelder

Für jeden abstimmenden Tensor muss man ein Votierfeld an seine Position verschieben und dann in seine Haupttrichtung drehen, d. h. in Richtung des Eigenvektors \hat{e}_1 des Tensors, also z. B. in Tangentenrichtung eines Kurvenelements. Dies ist notwendig, weil das Votierfeld von einem waagrechten, zentralen Kantenelement ausgeht und fächerartig die bestmöglichen Verbindungen zu Nachbarelementen darstellt. Ist ein Bildmerkmal (das zentrale Kantenelement) gedreht, so muss auch das Votierfeld entsprechend gedreht werden (siehe Abbildung 2.5). Damit erhält man den Einfluss jedes abstimmenden Tensors auf seine Umgebung.

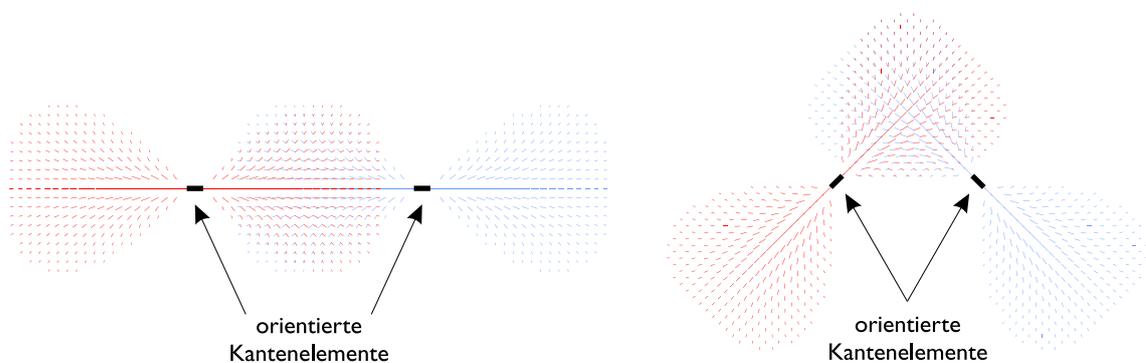


Abbildung 2.5: Überlappung zweier Stab-Votierfelder, die in einem Winkel von 0° und 90° zueinander stehen. Bei einem Winkel von 90° ist die Unsicherheit der Verbindungslinie am größten; sie kann von einer runden Kurve bis hin zu einer spitzen Ecke alles sein.

2 Das Tensor Voting Verfahren

Den Wert an einer Stelle p eines gedrehten Votierfeldes berechnet man folgendermaßen:

$$V_{\text{rot}}(p) = R_{\psi\phi\theta} V(R_{\psi\phi\theta}^{-1} p) R_{\psi\phi\theta}^T \quad (2.5)$$

wobei V_{rot} das gedrehte und V das im Voraus berechnete Votierfeld ist. $R_{\psi\phi\theta}$ ist die entsprechende Rotationsmatrix.

2.4.2 Votierung

Die im vorangegangenen Schritt erhaltenen Einflussbereiche überlagern sich mehr oder weniger und müssen nun zusammengefasst werden. Dies geschieht, indem man an jeder Stelle alle *Stimmen*, d. h. Tensoren in den ausgerichteten Votierfeldern, sammelt und auf den empfangenden Tensor aufaddiert. Man unterscheidet dabei zwei Arten der Tensorvotierung, die in den folgenden beiden Abschnitten erklärt werden.

Spärliche Tensorvotierung

Bei der sogenannten *spärlichen* Tensorvotierung erhalten nur diejenigen Stellen in der Tensorkarte Stimmen, die ein Eingabemerkmale aufweisen. Als Ergebnis erhält man eine spärliche Tensorkarte. Wie bereits oben beschrieben arbeitet man beim Votieren mit den einzelnen Komponenten eines Tensors, um den Vorgang zu vereinfachen und zu beschleunigen. Man muss also die Anteile der Stab- und Ballkomponente getrennt aufaddieren und erhält folgende Formel für die spärliche Votierung:

$$\mathbf{T}_i^{\text{neu}} = \begin{cases} \mathbf{0}, & \mathbf{T}_i^{\text{alt}} = 0 \\ \mathbf{T}_i^{\text{alt}} + \sum_{k \in N(i)} \left((\lambda_{1,k} - \lambda_{2,k}) V_{k,i}^S + \lambda_{2,k} V_{k,i}^B \right), & \mathbf{T}_i^{\text{alt}} \neq 0 \end{cases} \quad (2.6)$$

wobei $N(i)$ die Indizes der Tensoren in der Nachbarschaft des i -ten Tensors zurückgibt. $V_{k,i}^S$ und $V_{k,i}^B$ sind die Stimmen des bereits verschobenen und rotierten Stab- bzw. Ball-Votierfeldes für den empfangenden Tensor i . $\lambda_{1,k}$ und $\lambda_{2,k}$ sind die Eigenwerte des abstimmenden, k -ten Tensors.

Dichte Tensorvotierung

Neben der spärlichen Votierung gibt es auch eine sogenannte *dichte* Tensorvotierung:

$$\mathbf{T}_i^{\text{neu}} = \mathbf{T}_i^{\text{alt}} + \sum_{k \in N(i)} \left((\lambda_{1,k} - \lambda_{2,k}) V_{k,i}^S + \lambda_{2,k} V_{k,i}^B \right) \quad (2.7)$$

Im Gegensatz zur spärlichen Votierung erhalten bei der dichten Votierung *alle* Stellen Stimmen bzw. alle, die sich im Einflussbereich irgendeines Eingabemerkmals befinden. Dieses Verfahren liefert eine dichte Tensorkarte.

In der Regel ist der erste Votierdurchgang eine spärliche Votierung, der zweite eine dichte. Die Votierung mit der Ballkomponente ist nur anfangs notwendig, um Informationen über die Orientierung der Tensoren zu gewinnen. Im zweiten Durchgang, wenn es um die Verbreitung der Richtungsinformation geht, votiert man nur noch mit der Stabkomponente.

2.4.3 Zerlegung jedes Tensors in sein Eigensystem

Nachdem alle Tensoren abgestimmt haben, muss man für alle Tensoren, die nicht mehr in diagonalisierter Form vorliegen, die Eigenwerte und Eigenvektoren neu berechnen, so dass man wieder die Form in Gleichung 2.2 bzw. die Komponentendarstellung erhält.

2.4.4 Erstellen der Salienzkarten

Am Ende des Votiervorgangs erhält man eine neue Tensorkarte, aus der man in diesem Schritt zwei sogenannte *Salienzkarten* berechnet (drei im dreidimensionalen Raum). Eine Salienzkarte spiegelt einzelne Teile der kompakten Information der Tensorkarte wider, d. h. sie gibt die Gewissheit für *ein* geometrisches Merkmal und dessen Beschreibung an jeder Stelle der Karte an. Die Karten werden als gerichtete Vektorfelder repräsentiert, in denen an jeder Stelle sowohl die Stärke s als auch die Richtung \hat{v} mit Einheitslänge bekannt sind. Im zweidimensionalen Raum gibt es folgende Salienzkarten (siehe [MLT00] für den dreidimensionalen Fall):

- *Kurvensalienzkarte (curve saliency map, CMap)*. $s = \lambda_1 - \lambda_2$, $\hat{v} = \hat{e}_1$ ist die Tangentenrichtung des Kurvenelements. Eine Kurve ragt heraus, d. h. hat eine große Salienz s , wenn sie viele Stimmen erhalten hat (großes λ_1) und deren Orientierung gleichmäßig ist (kleines λ_2).
- *Kreuzungssalienzkarte (junction saliency map, JMap)*. $s = \lambda_2$, \hat{v} kann willkürlich gewählt werden (Kreuzungen haben natürlich keine Richtung). Kreuzungspunkte haben eine geringe Exzentrizität ($\lambda_1 \approx \lambda_2$), weil sie Stimmen aus unterschiedlichen Richtungen erhalten haben.

Die Salienzkarten sind das eigentlich interessante Ergebnis des Votiervorgangs und dienen als Ausgangspunkt für die Extraktion der geometrischen Strukturen (siehe Abschnitt 2.6).

2.5 Votierfelder

Ein Votierfeld ist ein gerichtetes Tensorfeld, das den Einfluss eines einzelnen Eingabemerkmals, d. h. eines Punktes, Kurven- oder Flächenelements, auf seine Nachbarschaft beschreibt.² Jede Stelle des Votierfeldes wird durch eine Richtung und eine Stärke beschrieben. Diese Eigenschaften sind entscheidend für das Tensor Voting Verfahren, denn sie bestimmen, in welchem Maße geometrische Strukturen erkannt werden können. Sie geben die wahrscheinlichste Verbindungslinie zweier Eingabemerkmale vor.

Bei der Implementierung von Votierfeldern genügt es, ein Vektorfeld mit den Hauptrichtungen der Tensoren zu verwenden und erst bei Bedarf, d. h. bei der Aufsummierung im Votiervorgang, die höherrangigen Tensoren wieder zu rekonstruieren. Man wandelt einen Vektor $v = (v_x \ v_y)^T$ im Votierfeld durch das sogenannte *dyadische Produkt* in einen Tensor zweiter Ordnung um:

$$\mathbf{V} = vv^T = \begin{pmatrix} v_x^2 & v_x v_y \\ v_x v_y & v_y^2 \end{pmatrix} \quad (2.8)$$

wobei \mathbf{V} ein Tensor ist und jederzeit wieder in sein Eigensystem zerlegt werden kann, wodurch man die Form in Gleichung 2.2 erhält.

Votierfelder haben eine unendliche Ausdehnung, jedoch spielt das in der Praxis keine Rolle, da die Stärken mit der Entfernung abnehmen und ab einem gewissen Punkt vernachlässigbar gering werden – man kann also das Feld in seiner Ausdehnung begrenzen (siehe dazu Abschnitt 2.5.4). Außerdem kann man – anstatt die Stimme im Kontinuum zu berechnen – diskrete Votierfelder verwenden, die man im Voraus mit einer bestimmten Auflösung berechnet, wodurch dann der spätere Votierprozess etwas beschleunigt wird. In diesem Fall sollte man jedoch die Werte interpolieren (siehe dazu Abschnitt B.1), die man aus einem gedrehten Votierfeld liest, um die Qualität zu verbessern.

²Das Prinzip der Votierfelder ist nicht neu. Neumann und Mingolla beschreiben in [NM01] formale Ansätze für „Assoziationsfelder“ bei der Wahrnehmungsgruppierung (perceptual grouping) einschließlich psychophysischer und psychologischer Aspekte und bieten eine Übersicht über ältere und aktuelle neuronale Modelle.

2.5.1 Das grundlegende, zweidimensionale Votierfeld

Das zweidimensionale Stab-Votierfeld wird als *grundlegendes Votierfeld* bezeichnet, denn man kann alle übrigen Votierfelder von ihm ableiten. Es beschreibt den Einfluss eines Kurvenelements der Einheitslänge auf seine Nachbarschaft. Die nachfolgenden Abschnitte beschreiben, welche Kriterien beim Entwurf beachtet werden müssen, wie zwei Elemente zueinander in Beziehung stehen (in [NM01] *relatability* genannt) und wie der Einfluss räumlich gewichtet wird (*spatial weighting*).

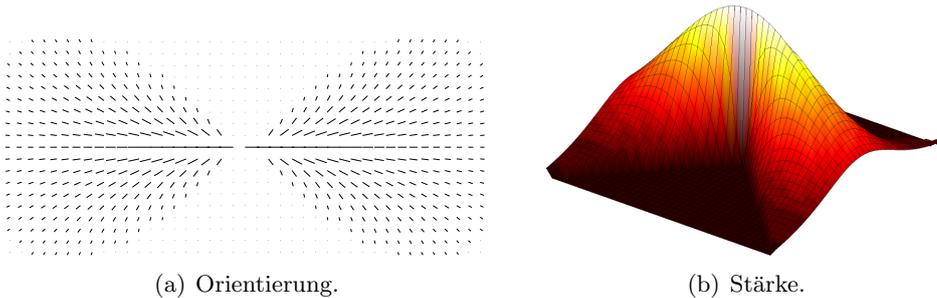


Abbildung 2.6: Das grundlegende, zweidimensionale Votierfeld.

Anforderungen

Wie bereits eingangs erwähnt sind die Eigenschaften des Votierfeldes ganz entscheidend für die Erkennungsleistung geometrischer Strukturen. Basierend auf den Prinzipien der Gestaltpsychologie (siehe z. B. [Wer23]) haben sich Guy und Medioni [GM96] deshalb an folgenden Konzepten der Wahrnehmung orientiert, die beim Entwurf des grundlegenden Votierfeldes berücksichtigt werden müssen:

- *Ko-Kurvilinearität.* Solange keine anderen Hinweise vorhanden sind, ist eine glatte Fortsetzung die einzige Interpretationsmöglichkeit.
- *Konstanz der Krümmung.* Die menschliche Wahrnehmung neigt dazu, Kurven mit einer bestimmten Krümmung ein Stück weit mit derselben Krümmung zu verlängern.
- *Bevorzugung geringerer Krümmungen gegenüber größeren.* Wir scheinen unterbrochene Liniensegmente so zu verbinden, dass die Totalkrümmung (siehe unten) minimal wird.
- *Nähe.* Nahe beieinander liegende Bereiche beeinflussen sich stärker als weiter entfernt liegende.

Orientierung

Der erste Aspekt beim Entwurf des grundlegenden Votierfeldes ist die Beziehung zwischen zwei Merkmalen P und Q . Da geringe und konstante Krümmungen bevorzugt werden sollen, ergibt sich die Orientierung an einem Punkt Q des Votierfeldes aus der Tangente des Kreises, der durch den Ursprung P und den Punkt Q geht (siehe Abbildung 2.7). Bei Punkten auf der x -Achse entartet der Kreis zu einer geraden Linie.

Als Verbindungslinie wird deshalb ein Kreis gewählt, weil dieser die *Totalkrümmung* (total curvature) wie oben gefordert minimiert. Die Totalkrümmung einer Kurve ist folgendermaßen

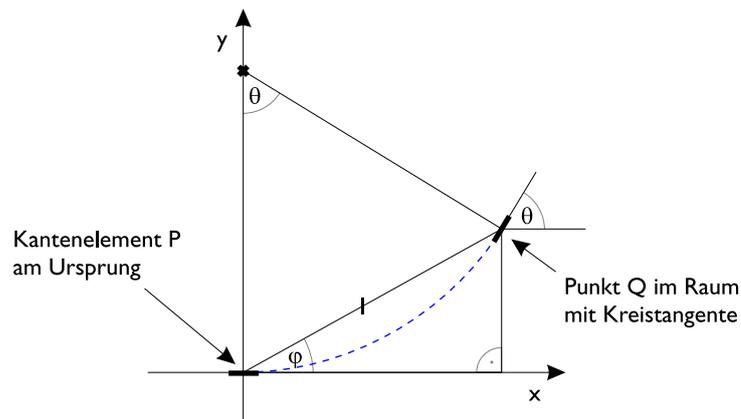


Abbildung 2.7: Ermitteln der Richtung für jeden Punkt im Raum.

definiert [MLT00]:

$$TC_{\alpha}(L) = \int_0^L |\rho(s)|^{\alpha} ds, \quad \alpha > 1 \quad (2.9)$$

wobei L die Länge der Kurve und $\rho(s)$ die Krümmung an einer Stelle s angibt. α ist ein frei wählbarer, skalarer Wert und wird üblicherweise mit 2 belegt. Guy und Medioni haben jedoch gezeigt, dass für alle Werte α größer 1 eine kreisförmige Verbindung die Totalkrümmung minimiert.

Man kann also die Orientierung an einer durch den Vektor $p = (x \ y)^T$ angegebenen Stelle wie folgt bestimmen:

$$R(p) = \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix} \quad \text{mit } \theta = 180^{\circ} - 2(90^{\circ} - \varphi) = 2\varphi \text{ und } \varphi = \tan^{-1} \left| \frac{y}{x} \right| \quad (2.10)$$

Räumliche Gewichtung

Der zweite Aspekt ist die räumliche Gewichtung, die den gegenseitigen Einfluss zweier Merkmale beschreibt. Sie wird durch eine analytische Funktion, eine Gauß-Funktion, angenähert und kann z. B. anhand der Physik des Energieflusses zwischen zwei Punkten hergeleitet werden [MLT00]. Man geht dabei von einem Teilchen P_s aus, das in alle Richtungen Energie abgibt, die sich auf kreisförmigen Bahnen ausbreitet und von jeder beliebigen Stelle P_r zurückverfolgt werden kann (siehe Abbildung 2.8).

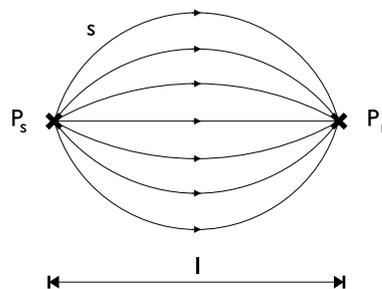


Abbildung 2.8: Energiefluss zwischen zwei Punkten.

Die Energiemenge, die auf einem bestimmten Pfad wandert, hängt zuerst einmal von dessen Länge s ab (und diese wiederum von der Entfernung l der beiden Punkte und der Anfangsrichtung φ des Pfades), denn auf dem direkten, kürzesten Weg fließt am meisten Energie. Außerdem

2 Das Tensor Voting Verfahren

bestimmt die Krümmung ρ eines Pfades die Energiemenge, da bei mittleren Entfernungen (bei denen die Krümmung gering ist) die Energie gleichmäßiger über die Pfade verteilt fließt. Diese Überlegungen führen zu der folgenden *Abschwächungsfunktion* (decay function):

$$DF(s, \rho) = e^{-\left(\frac{s^2 + c\rho^2}{\sigma_v^2}\right)} \quad \text{mit } s = \frac{\varphi l}{\sin \varphi}, \quad \rho = \frac{2 \sin \varphi}{l} \quad (2.11)$$

Der Parameter c muss einmalig zur Entwurfszeit festgelegt werden und bestimmt die Gewichtung der Kurvenkrümmung gegenüber der Entfernung der beiden Punkte. σ_v ist der sogenannte *Skalenfaktor*, der die Abschwächungsrate angibt (siehe Abschnitt 2.5.4).

Damit kennt man eine räumliche Gewichtung zweier Merkmale bezüglich Entfernung und Krümmung der Verbindungslinie. Allerdings gibt es noch eine Einschränkung bezüglich des Winkels φ . Wird dieser zu groß, d. h. größer als 45° , dann ist die kreisförmige Verbindung nicht mehr die beste, denn eine elliptische Verbindung beispielsweise hat in diesem Fall eine geringere Totalkrümmung, wie in Abbildung 2.9 zu sehen ist. Aus diesem Grund werden alle Stellen oberhalb und unterhalb der beiden Hauptdiagonalen auf Null gesetzt. Das bedeutet, dass wir für einen Punkt in diesem Bereich nicht votieren und dass zusätzliche Merkmale notwendig sind, um eine Verbindung zwischen einem solchen Paar herzustellen.

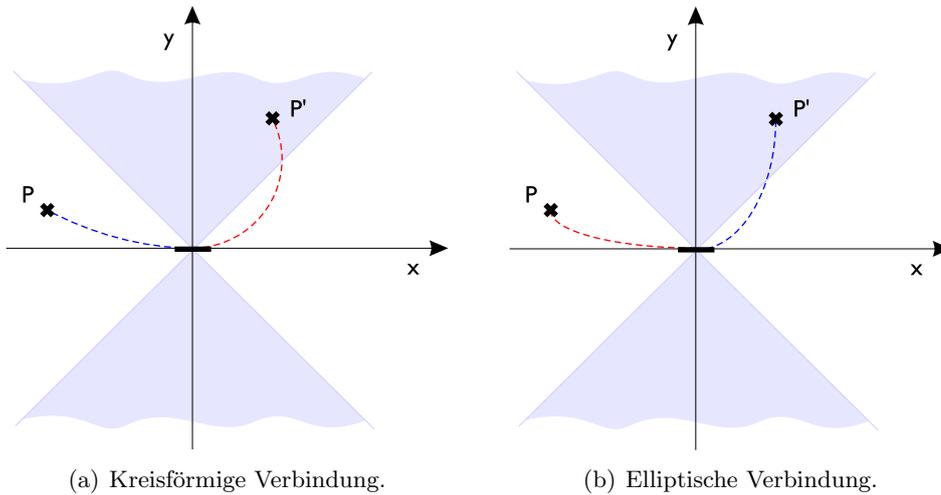


Abbildung 2.9: Innerhalb eines Bereichs von 90° (Punkt P unterhalb der Winkelhalbierenden) minimiert eine kreisförmige Verbindung (a) die Totalkrümmung. Außerhalb davon (Punkt P' oberhalb der Winkelhalbierenden) hat jedoch eine elliptische Verbindung (b) eine geringere Totalkrümmung.

Gesamtformel

Schließlich erhält man aus den Gleichungen 2.10 und 2.11 folgende Gesamtformel zur Konstruktion des grundlegenden Votierfeldes:

$$\tilde{V}^S(p) = \begin{cases} \mathbf{0}, & |y| > |x| \\ e^{-\left(\frac{s^2 + c\rho^2}{\sigma_v^2}\right)} \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix}, & |y| \leq |x| \end{cases} \quad (2.12)$$

wobei die Variablen s , ρ , θ , c und σ_v dieselbe Bedeutung haben wie zuvor. Der Vektor $p = (x \ y)^T$ gibt die Position im Votierfeld an. Außerdem ist zu beachten, dass $\tilde{V}^S(p)$ nur die Hauptrichtungen

der Tensoren enthält (zur Konstruktion der Tensoren siehe Gleichung 2.8). Die obige Gleichung lässt sich durch Eliminieren der aufwändigen trigonometrischen Funktionen noch optimieren zu:

$$\tilde{V}^S(p) = \begin{cases} \mathbf{0}, & |y| > |x| \\ e^{-\frac{x^2}{\sigma_v^2}} \begin{pmatrix} 1 \\ 0 \end{pmatrix}, & y = 0 \\ e^{-\left(\frac{s^2 + c \frac{1}{R^2}}{\sigma_v^2}\right)} \begin{pmatrix} 1 - \frac{y}{R} \\ \frac{x}{R} \end{pmatrix}, & |y| \leq |x| \end{cases} \quad (2.13)$$

mit

$$s = \frac{\varphi(x^2 + y^2)}{y} \quad \text{und} \quad R(x, y) = \frac{x^2 + y^2}{2y}$$

2.5.2 Das zweidimensionale Ball-Votierfeld

Das zweidimensionale Ball-Votierfeld definiert den Einfluss eines richtungsunabhängigen (isotropen) Tensors auf seine Umgebung. Im Gegensatz zum grundlegenden Votierfeld ist es daher gleichartig in jede Richtung aufgebaut (siehe Abbildung 2.10).

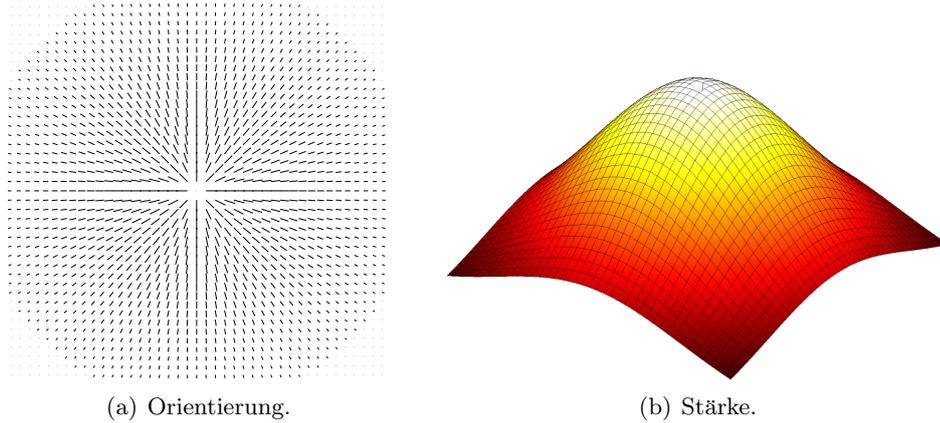


Abbildung 2.10: Das zweidimensionale Ball-Votierfeld.

Medioni et al. [MLT00] leiten das zweidimensionale Ball-Votierfeld aus dem grundlegenden Votierfeld ab, indem sie dieses einmal um die z -Achse drehen. So gilt für jede durch den Vektor p beschriebene Stelle:

$$V^B(p) = \int_0^\pi R_{\psi\phi\theta} V^S(R_{\psi\phi\theta}^{-1} p) R_{\psi\phi\theta}^T d\theta \Big|_{\psi=0, \phi=0} \quad (2.14)$$

wobei $R_{\psi\phi\theta}$ die Rotationsmatrix mit den Winkeln ψ , ϕ und θ um die x -, y - und z -Achse ist. Implementieren lässt sich das Votierfeld jedoch wesentlich einfacher, indem man die Abschwächungsfunktion $DF(s, \rho)$ (siehe Gleichung 2.11) mit dem entsprechenden Richtungsvektor multipliziert:

$$\begin{aligned} \tilde{V}^B(p) &= \begin{cases} \mathbf{0}, & p = 0 \\ DF(\|p\|, 0) \cdot \frac{p}{\|p\|}, & p \neq 0 \end{cases} \\ &= \begin{cases} \mathbf{0}, & p = 0 \\ e^{-\frac{\|p\|^2}{\sigma_v^2}} \frac{p}{\|p\|}, & p \neq 0 \end{cases} \end{aligned} \quad (2.15)$$

2 Das Tensor Voting Verfahren

Beim Ball-Votierfeld ist zu beachten, dass der Mittelpunkt ungerichtet ist und eine Stärke von 0 hat, da es keine Vorzugsrichtung gibt.

2.5.3 Dreidimensionale Votierfelder

Die Votierfelder, die für das Tensor Voting Verfahren im dreidimensionalen Raum benötigt werden, können ebenso wie das zweidimensionale Ball-Votierfeld durch eine oder mehrere Rotationen aus dem grundlegenden Votierfeld erzeugt werden. Hierbei sei jedoch auf die weiterführende Literatur [MLT00], [Lee98] und [Tan00] verwiesen.

2.5.4 Der Skalenfaktor

Die Votierfelder werden nur durch einen frei wählbaren Parameter festgelegt, den Skalenfaktor σ_v (ein typischer Wert für σ_v ist 18.25). Der Skalenfaktor legt die Entfernung fest, bis zu der ein Element Einfluss auf andere Elemente ausübt. Er wirkt sich auf die Erkennungsleistung des Verfahrens bezüglich unterschiedlicher Bildstrukturen aus. Beispielsweise kann man mit einem großen Skalenfaktor noch grobe Strukturen erkennen, die durch weit voneinander entfernte Eingabemerkmale angedeutet werden, während man einen kleinen Skalenfaktor wählen muss, um feine Strukturen auflösen zu können. Näheres zu Skalenproblemen und Skalenräumen ist beispielsweise in [Neu02b] und [Lin94] und in Abschnitt 7.4.1 zu finden.

Man kann bei der Konstruktion eines diskreten Votierfeldes die Größe automatisch bestimmen, indem man dafür den Wert $3\sigma_v$ annimmt. Dieser Wert ist aus der Stochastik bekannt und deckt bei der Normalverteilung – hier die Abschwächungsfunktion $DF(s, \rho)$ – einen nahezu hundertprozentigen Bereich ab (siehe z. B. [Fis89]).

Gruppierungsfähigkeit

Eine der Kernkompetenzen des Tensor Voting Verfahrens ist die Fähigkeit, geometrische Merkmale zu größeren Strukturen zusammenzufassen. Wie bereits oben angedeutet hängt sie direkt vom Skalenfaktor ab, weil dieser die Größe des gegenseitigen Einflussbereichs und damit der Votierfelder festlegt. Bei der dichten Votierung müssen sich die Votierfelder also zumindest über die halbe Distanz zweier zu überbrückender Merkmale ausdehnen, so dass sie aneinander stoßen oder – noch besser – sich überlappen und so genügend große Salienzen auf der Verbindungslinie erzeugen können (siehe Abbildung 2.11). Bei der spärlichen Votierung dagegen müssen sie die gesamte Distanz überbrücken, um gegenseitig Einfluss ausüben zu können.

Manchmal kann es sinnvoll sein, das Verfahren mehrmals anzuwenden (vgl. Abschnitt 6.4), wenn das Ergebnis nach einmaliger Durchführung noch nicht zufriedenstellend ist und noch kleinere Lücken enthält (z. B. wegen Anwendung eines Schwellwertverfahrens).

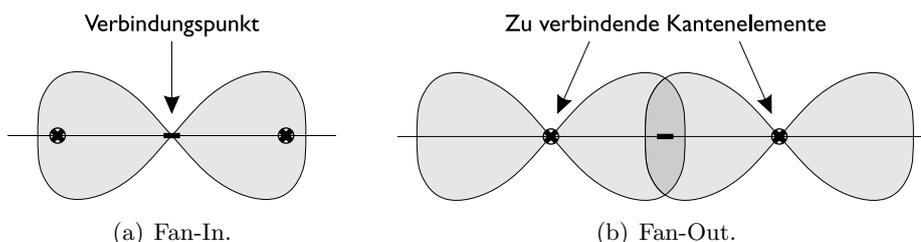


Abbildung 2.11: Unterschiedliche Perspektiven auf zwei zu überbrückende Kantenelemente: (a) aus der Sicht des Verbindungspunktes wird das Feld als *Fan-In*, (b) aus der Sicht der Kantenelemente als *Fan-Out* bezeichnet.



Abbildung 2.12: Das Kanisza-Dreieck (a) veranschaulicht das Prinzip der illusionären Konturen. (b) und (c) zeigen die Ergebnisse nach einmaliger bzw. zweimaliger Anwendung des Tensor Voting Verfahrens.

Unterbrochene Konturen. Obwohl das Tensor Voting Verfahren sehr gut mit Lücken in Linien und Kurven zurechtkommt, kann dabei das Problem auftreten, dass die Fortführung von Kurven zu wenig deren Krümmung berücksichtigt, dass beispielsweise eine fehlende Seite eines Kreises durch eine eher spitze Ecke anstatt eines runden Bogenstücks ersetzt wird. Abhilfe verspricht hier eine Erweiterung des Verfahrens durch Berücksichtigung der Krümmung (siehe Abschnitt 6.2).

Illusionäre Konturen. Unser visuelles System kann in bestimmten Bildern Konturen erkennen, die in Wirklichkeit gar nicht vorhanden, sondern nur angedeutet sind. Ein gutes Beispiel dafür sind Kanisza Figuren.³ Eine der bekanntesten Figuren ist das Kanisza Dreieck, das hier stellvertretend untersucht wurde. Abbildung 2.12 zeigt das Originalbild und die Ergebnisse nach ein bzw. zwei Anwendungen der Tensorvotierung (siehe Abschnitt 6.4 für eine ausführlichere Beschreibung mehrerer Anwendungen der Tensorvotierung). Je nach Skala kann man die Kontur des illusionären Dreiecks bzw. der Eckpunkte extrahieren (z. B. $\sigma_v = 50$ bzw. $\sigma_v = 3$). Das Auffinden illusionärer Konturen kann allerdings auch ein Nachteil sein, wenn man die tatsächlich vorhandenen Konturen extrahieren möchte, das Verfahren jedoch eine nicht vorhandene Kontur findet. Bei den Herzbildern haben illusionäre Konturen jedoch keine Bedeutung.

2.6 Extraktion geometrischer Strukturen

Am Ende des Tensor Voting Verfahrens steht der Prozess zur Merkmalsextraktion, der als Eingabe die Salienzkarten bekommt und daraus geometrische Strukturen (Kreuzungspunkte, Kurven, Regionen) gewinnt. Da jede Struktur andere Eigenschaften hat, muss man für deren Extraktion unterschiedliche Verfahren anwenden.

Kreuzungspunkte lassen sich relativ einfach ermitteln, indem man in der Kreuzungssalienzkarte lokale Maxima herausarbeitet, beispielsweise indem man den Intensitätswert jedes Bildpunktes mit seiner Achter-Nachbarschaft⁴ vergleicht und ihn nur dann erhält, wenn er größer als die Werte aller Nachbarn ist.

Zur Extraktion von Kurven muss man anders vorgehen, da man ja möglichst zusammenhängende Konturen und nicht einzelne Punkte erhalten will. Dafür bietet sich z. B. die nachfolgend beschriebene Non-maximum Suppression Methode an.

2.6.1 Die Non-maximum Suppression Methode

Die Non-Maximum Suppression Methode nach Canny [Can83, Rot01] wird als Verarbeitungsschritt zur Kantendetektion verwendet und unterdrückt alle Bildpunkte, die keine lokalen Maxima in Richtung des Grauwertgradienten sind (hier: der Normalenvektor an einem Kurvenpunkt

³Nach Gaetano Kanisza, einem italienischen Psychologen.

⁴Die acht direkt angrenzenden Pixel in einem quadratischen Raster. Entsprechend gibt es auch eine Vierer-Nachbarschaft.

in der Salienzkarte). Die Kantenbereiche im Bild sind anfangs noch mehrere Pixel breit und durch unterschiedliche Grauwerte charakterisiert. Es wird angenommen, dass die eigentliche Kante im Maximum dieses Grauwertbereichs liegt, was bedeutet, dass alle anderen Grauwerte unterdrückt werden müssen. Dabei ist zu beachten, dass die Grauwerte entlang der Kante gleich sein können und das Verfahren deshalb nur in Richtung des Normalenvektors, d. h. orthogonal zur Kante, angewandt werden darf.

Mit Hilfe des normierten Normalenvektors \hat{n} an einem Kurvenpunkt p bestimmt man die beiden angrenzenden Punkte $p_l := p - \hat{n}$ und $p_r := p + \hat{n}$ und deren Grauwerte. Diese beiden Grauwerte müssen interpoliert werden (z. B. bilinear, siehe Abschnitt B.1), weil die Koordinaten von p_l und p_r in aller Regel reellwertig sind, während das Bild diskret ist. Sind nun die Grauwerte $g(p)$, $g(p_l)$ und $g(p_r)$ dieser drei Punkte gegeben, dann ist p ein lokales Maximum, falls gilt: $g(p) > g(p_l)$ und $g(p) > g(p_r)$. Andernfalls wird der Grauwert unterdrückt, d. h. auf Null gesetzt. Folglich hat man alle Kanten auf eine Breite von einem Pixel reduziert.

2.6.2 Schwellwertverfahren

Eine Salienzkarte – eventuell schon durch die Non-maximum Suppression Methode gefiltert – enthält oftmals noch Störungen in Form von kleinen, unförmigen Strukturen, die allerdings nur eine geringe Stärke aufweisen. Die eigentlich zugrunde liegenden Strukturen dagegen haben eine hohe Salienz. Diesen Sachverhalt kann man in einem Schwellwertverfahren nutzen, um die Störungen zu unterdrücken. Dabei bleiben alle Werte erhalten, die oberhalb eines bestimmten Schwellwertes liegen, während die übrigen Werte unterdrückt, d. h. auf Null gesetzt, werden.

Das Schwellwertverfahren kann sehr gute Ergebnisse liefern, jedoch besteht die Schwierigkeit darin, einen geeigneten Schwellwert für die Trennung zu finden. Eine Möglichkeit dafür bieten Methoden der Histogramm basierten Schwellwertsuche (siehe [Neu02a, Ots79]), deren Ergebnisse für bestimmte Zwecke ausreichen. Die automatische Schwellwertsuche lässt sich allerdings nur schwer auf ein breiteres Spektrum von unterschiedlichen Eingabedaten verallgemeinern und wird deshalb oft auf eine spezielle Aufgabenstellung zugeschnitten.

2.7 Zeitkomplexität

Eine gute Übersicht über die Zeitkomplexität des Tensor Voting Verfahrens ist in [TM98] zu finden. Im zweidimensionalen Fall beträgt die Komplexität für die Faltung mit den Votierfeldern $O(mk^2)$, wobei m die Anzahl der Eingabemerkmale und k die maximale Ausdehnung der Votierfelder in einer Dimension ist. Bei der dichten Votierung gilt $m = n^2$, wobei n die maximale Dimension der Bildmatrix ist. An die Faltung schließt sich die Aufsummierung und Interpretation der Stimmen (Zerlegung der Tensoren in ihre Eigensysteme, je $O(1)$) an, was an jedem Punkt in der Tensorkarte durchgeführt werden muss und $O(n^2)$ Zeit in Anspruch nimmt. Die Extraktion von Kurven und Kreuzungspunkten kostet jeweils eine Zeit von $O(n^2)$. Insgesamt ergibt sich also eine Zeitkomplexität von $O(n^2k^2)$.

Tensor Voting auf Grauwertbildern

Ein Problem des Tensor Voting Verfahrens, so wie in Kapitel 2 beschrieben, ist, dass es nicht direkt auf unterster Ebene ansetzt und auf ein Bild angewendet werden kann. Das Verfahren kann nur mit bereits extrahierten Eingabemerkmale (also mit Punkten, Kurven- und eventuell Flächenelementen) umgehen. Will man nun aber mit Bildern – oder vielmehr Grauwertbildern – arbeiten, so muss man zuerst einen Vorverarbeitungsschritt durchführen, der diese Eingabemerkmale gewinnt.

Dieses Kapitel zeigt die Problematik auf, die mit einfacheren Methoden der Merkmalsextraktion einhergeht, und schlägt weiterentwickelte Verfahren vor, die diese Probleme zu lösen versuchen. Eine Übersicht über verschiedene Verfahren der Merkmalsextraktion ist z. B. in [Joh01] und [Soj02] zu finden.

3.1 Ansatz

Damit ein Verfahren der Merkmalsextraktion zur Vorverarbeitung für das Tensor Voting Verfahren geeignet ist, muss es folgende Informationen als Ergebnis bereitstellen:

- *Eckpunkte.* Eine möglichst genaue Position (x, y) sollte für jeden Eckpunkt gefunden werden.
- *Kanten und deren Richtung.* Sowohl der Ort (x, y) als auch die Tangentenrichtung \hat{t} muss für jedes Kantenelement bekannt sein.
- *Salienzen.* Es ist wünschenswert, eine Stab- und Ball-Salienz für jedes geometrische Merkmal zu kennen, um mit Unsicherheiten umgehen zu können.

Die meisten Verfahren nutzen zur Erkennung dieser Merkmale direkt oder indirekt den Grauwertgradienten oder höhere Ableitungen der Grauwertfunktion. Dies ist leicht nachvollziehbar, denn eine Kante beispielsweise stellt einen Hell-Dunkel-Kontrast dar (siehe Abbildung 1.4) und bewirkt einen mehr oder weniger steilen Anstieg oder Abfall der Grauwertfunktion, der durch die Ableitung erfasst wird. Die Gradienten werden mittels einer Faltung der Bildfunktion mit einer speziellen Maske berechnet – meistens nach Sobel [GW01] oder mit einer Ableitung der Gauß-Funktion zur gleichzeitigen Glättung. Ein relativ neues Verfahren dagegen, das ohne Berechnung der Gradienten arbeitet, ist SUSAN [SB97].

3.2 Einfache Merkmalsextraktion

Die einfache Merkmalsextraktion konzentriert sich auf die Gewinnung einzelner Merkmalsarten, d.h. entweder von Ecken oder von Kanten. Die beiden Kategorien werden in den folgenden Abschnitten einzeln betrachtet.

3.2.1 Kantendetektion

Die einfachsten Verfahren zur Kantendetektion [Neu02a] nutzen direkt den Grauwertgradienten, der nach Robert, Prewitt oder Sobel berechnet wird, und wenden dann ein Schwellwertverfahren an. Diese Verfahren sind allerdings sehr anfällig gegenüber Störsignalen im Bild, da diese durch Ableitung verstärkt werden. Etwas ausgefeilter dagegen ist das Verfahren, das den Laplace von Gauß (LoG) Operator zur Kontrastdetektion verwendet, denn dieser setzt sich aus den partiellen Ableitungen zweiter Ordnung der Gauß'schen Dichtefunktion zusammen, was zu einer Glättung führt und dem Rauschen im Bild entgegenwirkt. Kontraste werden durch Nulldurchgänge (*zero-crossings*) in den Laplace-Antworten erkannt.

Als drittes, sehr häufig angewendetes Verfahren sei der Canny Kantendetektor [Can83] vorgestellt. Der Canny-Operator wurde als optimaler Kantendetektor entworfen und arbeitet in mehreren Phasen. Als erstes wird das Originalbild mit einem Gauß'schen Tiefpassfilter geglättet und dann werden die Richtungsableitungen berechnet, wodurch die Kontraste im Bild hervorgehoben werden. Der nächste Schritt besteht darin, Kanten bzw. Grate durch die Non-Maximum Suppression Methode freizustellen (siehe Abschnitt 2.6.1). Als letztes wird ein adaptives Schwellwertverfahren (*hysteresis thresholding*) mit zwei Schwellwerten T_u und T_o durchgeführt: Man wandert auf einem Grat, beginnend an einem Punkt größer T_o , in beide Richtungen solange, bis der Grat unter die untere Schwelle T_u fällt. Das durchwanderte Gratsegment bleibt dann als Kante erhalten. Durch das Schwellwertverfahren werden verrauschte Kanten aufgeteilt.

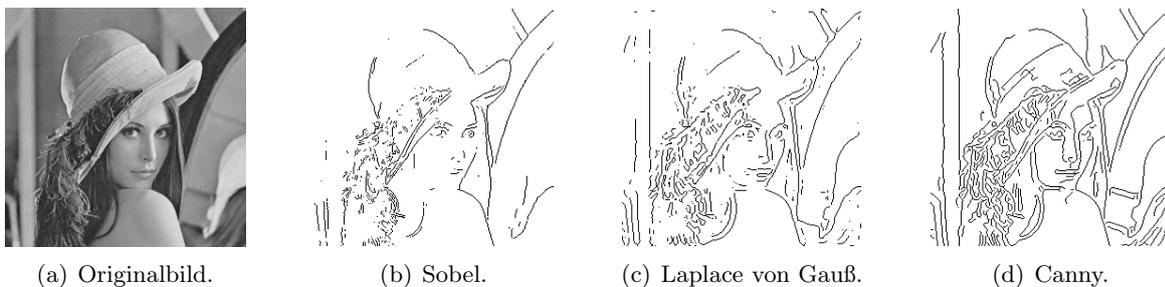


Abbildung 3.1: Vergleich verschiedener Kantendetektoren.

3.2.2 Eckendetektion

Einer der ältesten Eckendetektoren ist Beaudets [Bea78]. Er verwendet die Determinante der Hesse-Matrix H (Definition siehe Gleichung 3.1) zur Klassifikation einzelner Pixel. Ein Punkt, an dem die Determinante positiv und größer als ein bestimmter Schwellwert ist, wird als Eckpunkt markiert. Die Determinante ist mit der *Gauß'schen Krümmung* verbunden, die über die beiden Hauptkrümmungen κ_1 und κ_2 der Oberfläche $z = g(x, y)$ an dem zu betrachtenden Punkt definiert ist [Soj02]:

$$K = \kappa_1 \kappa_2 = \frac{\det H}{(1 + \|\nabla g\|^2)^2} = \frac{g_{xx}g_{yy} - g_{xy}^2}{(1 + g_x^2 + g_y^2)^2} \quad \text{mit } H := \begin{pmatrix} g_{xx} & g_{xy} \\ g_{xy} & g_{yy} \end{pmatrix} \quad (3.1)$$

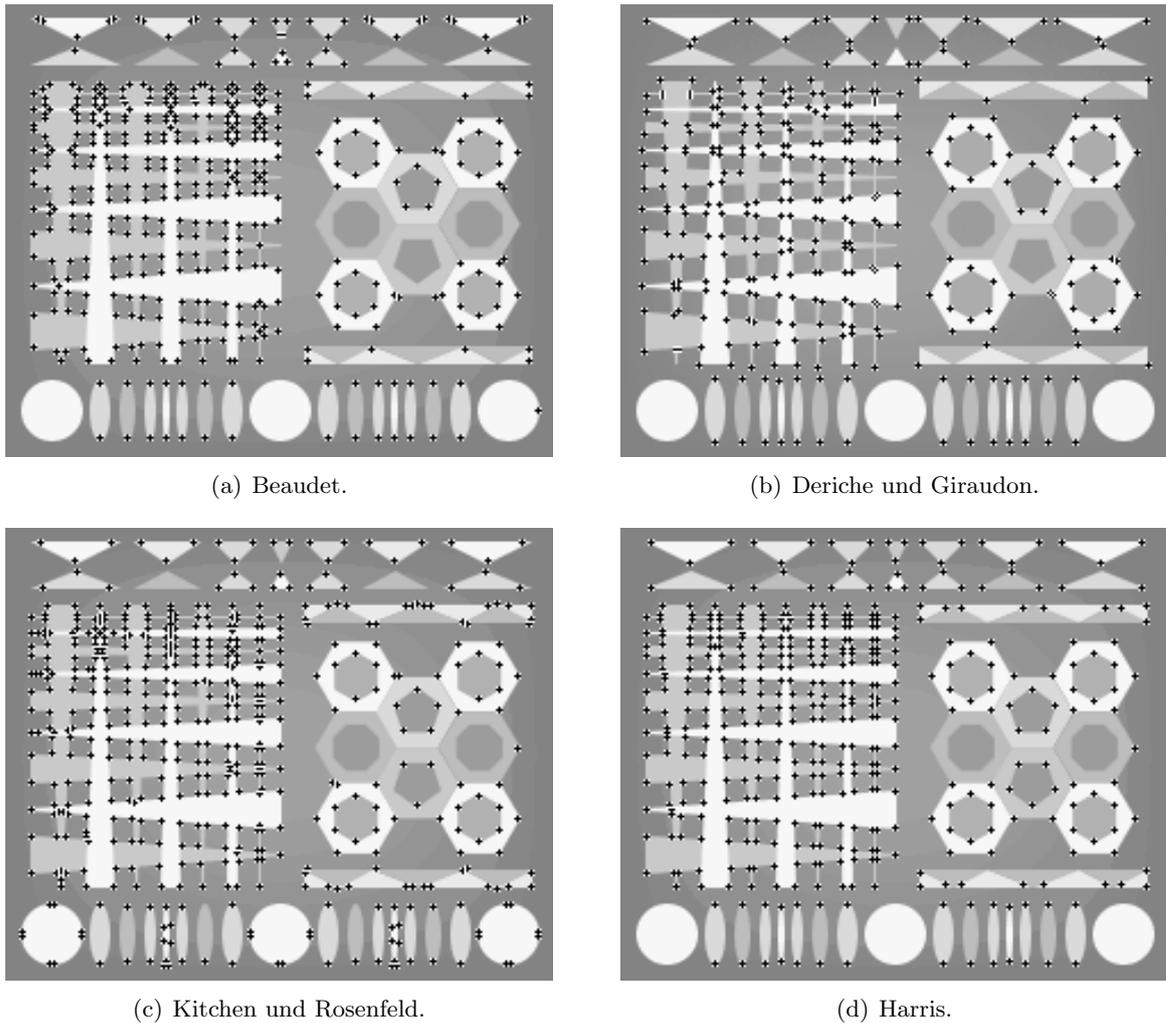


Abbildung 3.2: Vergleich verschiedener Eckendetektoren. Der Beaudet-Operator (a) lokalisiert die Eckpunkte stets etwas „innerhalb“ der Ecken, während der Detektor von Deriche und Giraudon (b) zwar die Position der Eckpunkte besser findet, dafür aber weniger erkennt. Der Detektor von Kitchen und Rosenfeld (c) erkennt mehr Ecken, findet allerdings auch einige falsch positive. Das beste Ergebnis liefert hier der Harris-Detektor (d). (Die Bilder stammen von Eduard Sojka [Soj03] und sind mit Erlaubnis wiedergegeben. Zur Verdeutlichung wurden gegenüber den Originalen die Eckpunkte hervorgehoben und die Hintergründe etwas aufgehellt.)

Mit Hilfe der Gauß'schen Krümmung kann man jeden Bildpunkt in eine von drei Kategorien einteilen: für die Bereiche $K > 0$, $K = 0$ und $K < 0$ gilt die Einteilung in *elliptisch*, *parabolisch* und *hyperbolisch*. Das bedeutet also, dass der Beaudet-Operator elliptische Punkte erkennt, bei denen die Gauß'sche Krümmung ein lokales Maximum hat. Darin liegt jedoch ein Problem, denn die erkannten Eckpunkte liegen immer etwas „innerhalb“ einer Ecke. Dreschler und Nagel [DN82] haben deshalb ein verbessertes Verfahren entwickelt, das die lokalen Extremwerte der Gauß'schen Krümmung ermittelt und dann Paare von elliptischen und hyperbolischen Punkten in unmittelbarer Nähe sucht. Ein Eckpunkt liegt dann auf der Linie zwischen den beiden Extrema an der Stelle, an der die Gauß'sche Krümmung gleich Null ist.

Es gibt noch eine ganze Reihe weiterer Verfahren (siehe [Soj02] und [Joh01]), z. B. von Kitchen und Rosenfeld, die die Änderung der Gradientenrichtung als Kriterium heranziehen; von Deriche und Giraudon, die den Beaudet-Operator auf zwei unterschiedlichen Skalen anwenden,

um Eckpunkte besser lokalisieren zu können; von Harris und Stephens¹, die einen Strukturtensor (siehe Abschnitt 3.3.1) verwenden; und von Noble, dessen Verfahren mit dem Harris-Detektor verwandt ist.

3.2.3 Probleme

Das Hauptproblem bei den Verfahren der einfachen Merkmalsextraktion ist, dass man nur eine Merkmalsart (Kanten oder Ecken) erhält. Benutzt man nur Ecken zur Weiterverarbeitung, so fehlt jegliche Richtungsinformation, benutzt man nur Kanten, so geht die Information über Kreuzungspunkte verloren. Um Abhilfe zu schaffen, könnte man zwei Verfahren kombinieren. Das würde allerdings einen unnötigen Mehraufwand bedeuten und könnte zu einer Inkonsistenz der Ergebnisse der beiden Verfahren führen, wenn sie nicht harmonieren, z. B. hinsichtlich der Skala oder des Glättungsfaktors. Hinzu kommen die einzustellenden Parameter, die man der Einfachheit halber möglichst gering halten will, jedoch benötigt beispielsweise alleine der Canny-Operator schon drei Parameter (den Glättungsfaktor und zwei Schwellwerte). Wünschenswert wäre darüber hinaus, eine Salienz für jedes Merkmal zu finden, d. h. ein Maß über Punkt- und Kurvenartigkeit, um mit Unsicherheiten umgehen zu können und nicht streng in zwei Kategorien einteilen zu müssen.

Bei Kantendetektoren ist noch eine weitere Besonderheit zu beachten: Viele liefern als Ergebnis nicht direkt Kantenelemente, zu denen man die Richtung kennt, sondern eine Karte mit den extrahierten Kantenpunkten. Um daraus dann die notwendige Richtungsinformation zu erhalten, kann man nicht einfach den Gradienten dieser Karte berechnen, weil dieser bei den Linien zwei Ausschläge liefern würde (einmal für den Anstieg und einmal für den Abfall der Kante), sondern man bräuchte eine ausgefeiltere Methode.

3.3 Polymorphe Merkmalsextraktion

Im Gegensatz zur einfachen Merkmalsextraktion werden bei der *polymorphen Merkmalsextraktion* [FH98] gleichzeitig mehrere Merkmalsarten (Punkte, Kanten und eventuell Regionen) gewonnen. Das hat den entscheidenden Vorteil, dass nur ein Verfahren notwendig ist und dass keine Strategien zur Konfliktlösung entwickelt werden müssen. Insgesamt sollen durch diesen Ansatz die oben geschilderten Probleme behoben werden. Eine sehr wichtige Idee hierfür ist der Strukturtensor, der – wie der Name bereits sagt – etwas über Struktur und Orientierung an jeder Position im Bild aussagt. Darauf aufbauend gibt es eine Methode von Förstner [För94], die lokale, statistische Eigenschaften der Bildfunktion ausnutzt, um Merkmale zu erkennen und Schwellwerte zu bestimmen. Eine andere Idee ist die Anwendung von Gabor-Filtern, die ganz gezielt Strukturen in einer bestimmten Orientierung und Größe erkennen können.

3.3.1 Der Strukturtensor

Das Ziel des Strukturensors ist, die Geometrie in einem Bild durch die lokale Auswertung des Verlaufs der Grauwertfunktion möglichst gut zu beschreiben [Neu02a]. Für die Schätzung von vorherrschenden Richtungen wird der gemittelte Grauwertgradient in einer lokalen Umgebung $N(p)$ eines Punktes p verwendet. Die Mittelung des Grauwertgradienten erreicht man durch einen Boxfilter² oder Gauß'schen Tiefpassfilter, bezeichnet durch W_σ , wobei σ der Glättungsfaktor

¹Auch bekannt als Plessey-Detektor.

²Einer der einfachsten Tiefpassfilter; jeder Eintrag der Filtermaske ist gleich gewichtet, z. B. eine (3, 3)-Matrix, deren Einträge jeweils auf $\frac{1}{9}$ gesetzt sind [Neu02a].

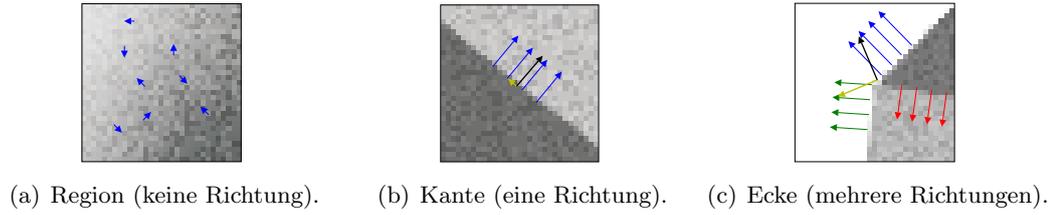


Abbildung 3.3: Gradientenrichtungen zur Berechnung des Struktur tensors. Die beiden Eigenvektoren des Struktur tensors sind schwarz (die Hauptrichtung) und gelb, jeweils fett gedruckt eingezeichnet.

ist. Der Struktur tensor $\mathbf{J}^\sigma(p)$ ergibt sich also aus einer Faltung des Glättungsfilters mit dem dyadischen Produkt des Gradienten:

$$\mathbf{J}^\sigma(p) = W_\sigma * (\nabla g(p) \cdot \nabla g(p)^T) = \begin{pmatrix} J_{xx} & J_{xy} \\ J_{xy} & J_{yy} \end{pmatrix} \quad (3.2)$$

mit

$$\begin{aligned} J_{xx} &= \int g_x^2(q) W_\sigma(p-q) \, dq && \approx \sum_{q \in N(p)} g_x^2(q) W_\sigma(p-q) \\ J_{xy} &= \int g_x(q) g_y(q) W_\sigma(p-q) \, dq && \approx \sum_{q \in N(p)} g_x(q) g_y(q) W_\sigma(p-q) \\ J_{yy} &= \int g_y^2(q) W_\sigma(p-q) \, dq && \approx \sum_{q \in N(p)} g_y^2(q) W_\sigma(p-q) \end{aligned}$$

Der Tensor $\mathbf{J}^\sigma(p)$ kann anschließend nach Gleichung 2.2 in sein Eigensystem zerlegt werden, wodurch man die Eigenwerte und Eigenvektoren erhält. An ihnen kann man Eigenschaften über die Struktur ablesen – entsprechend dem Tensor Voting Verfahren (siehe Abschnitt 2.4.4). Anhand der beiden Eigenwerte λ_1 und λ_2 , wobei λ_1 der größere von beiden ist, kann man folgende drei Kategorien unterscheiden:

- *Ecken.* $\lambda_1 > 0$, $\lambda_2 > 0$ mit der Kreuzungssalienz $s = \lambda_2$.
- *Kanten.* $\lambda_1 > 0$, $\lambda_2 \approx 0$ mit der Kurvensalienz $s = \lambda_1 - \lambda_2$, wobei der Eigenvektor \hat{e}_1 die Tangentenrichtung des Kantenelements angibt.
- *Regionen.* $\lambda_1 \approx \lambda_2 \approx 0$.

Die Berechnung von Struktur tensors scheint ideal zur Vorverarbeitung geeignet zu sein, da sie an jeder Stelle sowohl Kurven- und Kreuzungssalienz als auch die vorherrschende Richtung in Form eines Tensors liefert. Man erhält also als Ergebnis eine Tensorkarte, die man direkt für das Tensor Voting Verfahren weiter verwenden kann, d. h. es entfällt der Schritt der Tensorkodierung. Auf die Berechnung der Struktur tensors haben nur eine frei wählbare Variable, d. h. der Glättungsfaktor σ , und die Art der Gradientenberechnung (Sobel, Ableitungen der Gauß-Funktion etc.) Einfluss, was bedeutet, dass das Verfahren relativ einfach anzuwenden und ziemlich schnell durchzuführen ist.

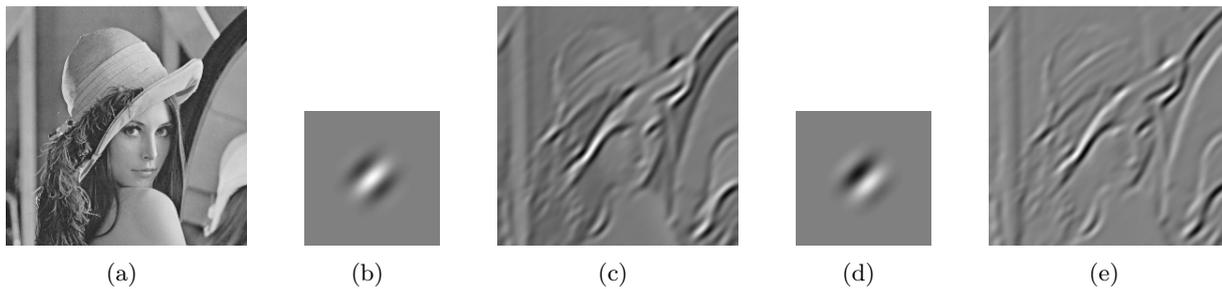


Abbildung 3.4: Anwendung eines geraden (b) und eines ungeraden (d) Gabor-Filters (Orientierung $\theta = 45^\circ$, 2 Zyklen, $\sigma = 5$) auf das Originalbild (a). Die jeweiligen Ergebnisse sind in (c) bzw. (e) zu sehen. Die gerade Filterkomponente wirkt als Liniendetektor, die ungerade als Kantendetektor.

3.3.2 Gabor-Filter

Einen ganz anderen Ansatz zur Merkmalsextraktion bietet die Filterung mit sogenannten *Gabor-Funktionen*. Sie zeichnen sich dadurch aus, dass sie die Unschärfe zwischen der Lokalisation von Ort und Frequenz minimieren, d. h. dass sie im Orts- und Frequenzbereich auf einen ähnlichen lokalen Wirkungsbereich beschränkt sind [Gáb46] (vgl. die Heisenberg'sche Unschärferelation). Die zuerst eindimensional definierten Gabor-Funktionen wurden dann später von Daugman [Dau88] auf den zweidimensionalen Fall verallgemeinert. Gabor-Funktionen können als Bandpassfilter (kurz: Gabor-Filter) zur selektiven Filterung orientierter Strukturen eingesetzt werden. Gabor-Filter haben außerdem einen biologischen Hintergrund: der primäre visuelle Kortex (von Säugetieren) führt eine ähnliche orientierungselektive Filterung durch [JP87], wodurch dieses Verfahren auch für technische Anwendungen sinnvoll erscheint.

Die zweidimensionale Gabor-Funktion ist das Produkt aus einer sinusförmigen, ebenen Welle mit einer Gauß'schen Trägerfunktion. Die Wellenfunktion setzt sich aus einem geraden und einem ungeraden Anteil (Kosinus bzw. Sinus) zusammen, was sich in einer komplexen Exponentialfunktion widerspiegelt:

$$f(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} e^{-2\pi i(u_0x+v_0y)} \quad (3.3)$$

Durch wenige Umformungsschritte erhält man eine Transferfunktion im Fourier-Raum, die der Originalfunktion sehr ähnlich ist:

$$F(u, v) = \frac{1}{2\pi\hat{\sigma}^2} e^{-\frac{(u-u_0)^2+(v-v_0)^2}{2\hat{\sigma}^2}} \quad \text{mit } \hat{\sigma} = \frac{1}{2\pi\sigma} \quad (3.4)$$

Die beiden Gleichungen enthalten drei Parameter: die Standardabweichung σ der Gauß-Funktion³ und die Koordinaten (u_0, v_0) , die einer Translation im Frequenzraum entsprechen. Anstatt kartesischer Koordinaten kann man auch polare verwenden und eine Distanz $r = \sqrt{u_0^2 + v_0^2}$ und einen Winkel $\theta = \tan^{-1} \frac{v_0}{u_0}$ angeben. Abbildung 3.4 zeigt ein Filter-Paar und die zugehörigen Ergebnisse nach deren Anwendung.

Will man Gabor-Filter zur Merkmalsextraktion nutzen, so bedarf es einer ganzen Familie von Filtern, die unterschiedlich ausgerichtet und eventuell für unterschiedliche Skalen ausgelegt sind (Parameter θ bzw. r). Aus diesem Satz an Filter-Antworten kann man dann die Stärke und Orientierung von geometrischen Merkmalen ermitteln. Um direkt Tensoren zweiter Ordnung zu

³Hier wird eine isotrope Trägerfunktion verwendet, d. h. derselbe σ -Wert sowohl für die x - als auch für die y -Richtung. Man kann auch eine nicht-isotrope Trägerfunktion wählen, die man dann durch eine geringfügige Änderung erhält.

erhalten, die als Eingabe für das Tensor Voting Verfahren dienen, kann man die in [Bab03] geschilderte Methode von Babós nutzen. Dabei stellt man zunächst für jede Orientierung θ der Filter einen sogenannten *Basis-Tensor* auf, der nichts anderes als ein in die entsprechende Richtung ausgerichtetes Stab-Tensor ist (vgl. Gleichung 2.8):

$$\mathbf{B}_\theta = b_\theta b_\theta^T = \begin{pmatrix} b_{\theta,x}^2 & b_{\theta,x}b_{\theta,y} \\ b_{\theta,x}b_{\theta,y} & b_{\theta,y}^2 \end{pmatrix} \quad \text{mit } b_\theta = \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix} \quad (3.5)$$

Anschließend gewichtet man die Basis-Tensoren entsprechend der Stärke a_θ der jeweiligen Filter-Antwort, die sich aus dem Betrag des komplexen Wertes der Antwort ergibt. Durch diese Art der Berechnung werden sowohl Kanten (durch den ungeraden oder imaginären Anteil) als auch Linien (durch den geraden oder reellwertigen Anteil) berücksichtigt.

$$\mathbf{T} = \sum_{\theta} a_\theta \mathbf{B}_\theta \quad (3.6)$$

Als letztes bildet man die Summe \mathbf{T} der Tensoren jeder Orientierung und zerlegt diese wie gewohnt (siehe Gleichung 2.2) in ihr Eigensystem, damit der resultierende Tensor einfach interpretiert und weiterverarbeitet werden kann. Durch die Gewichtung und Aufsummierung der Tensoren erhält man die gewünschten Salienzen für Kreuzungspunkte (Antworten aus mehreren Filtern) und Kurven (Antworten aus einem oder zwei benachbarten Filtern), d. h. eine Sicherheit dafür, ob es sich bei einem Merkmal um eine Ecke, um eine Kante oder um eine Region handelt.

3.3.3 Probleme

Da zur Berechnung der Strukturtenoren die Grauwertgradienten geglättet werden müssen, gehen unweigerlich Details im Bild verloren. Eine geringere Glättung verschlechtert allerdings die Qualität der Schätzung der vorherrschenden Richtungen. Deshalb muss man zwischen diesen beiden Kriterien abwägen und den Glättungsfaktor entsprechend wählen. Im Zweifelsfall sollte man für ihn einen kleinen Wert verwenden und damit Details erhalten, da man bei der Tensorvotierung die Richtungsschätzungen wieder verbessern, verloren gegangene Feinheiten jedoch nicht wiederherstellen kann.

Nicht gerade einfach bei dem beschriebenen Gabor-Verfahren ist die Einstellung der Parameter u_0 , v_0 und σ , weil die Gabor-Filter sehr gezielt auf Strukturen reagieren. Da man jedoch eine ganze Familie von Filtern verwendet, verliert diese Einstellung etwas an Wichtigkeit.

3.4 Ergebnisse

Für eine ausführliche Diskussion der Anwendung des Tensor Voting Verfahrens auf Grauwertbilder sei auf Babós [Bab03] verwiesen. Neben dem Strukturtenor- und Gabor-Verfahren wird darin noch ein weiteres, ähnliches Verfahren von Granlund und Knutsson vorgestellt.

Abbildung 3.5 zeigt ein Beispiel und die Ergebnisse für das Strukturtenor- und Gabor-Verfahren. In einigen einfachen Versuchen, in denen Strukturen wie ein Halbmond, ein Schachbrettmuster oder das Lena-Bild (siehe Abbildung 3.1(a)) untersucht wurden, unterschied das Gabor-Verfahren etwas schlechter als der Strukturtenor zwischen gerichteten und ungerichteten Strukturen, d. h. die Tensoren vieler gerichteter Strukturen hatten eine geringe Ball-Salienz. Dies hängt vermutlich mit der Art der Kombination der unterschiedlichen Filter-Antworten zusammen.

Beide hier vorgestellten Verfahren liefern dichte Tensorkarten, d. h. eine Vervollständigung von Merkmalen durch die Tensorvotierung ist teils gar nicht mehr notwendig. Vielmehr verlagert sich der Schwerpunkt auf die Glättung von Kurven und das Beseitigen von Störungen. Außerdem

entfällt der erste Durchgang der Tensorvotierung, weil die beiden Verfahren bereits gerichtete Strukturen, d. h. Stabkomponenten, ermittelt haben.

Für die Vorverarbeitungsstufe wurde in dieser Arbeit das Strukturtensor-Verfahren gewählt, weil es sich durch die direkte Berechnung der Tensoren sehr gut dafür eignet und es einfach umzusetzen ist.

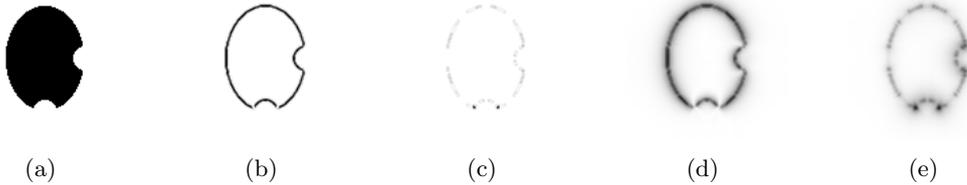


Abbildung 3.5: Salienzkarten für Struktur- und Gabor-Tensoren. (a) zeigt das Eingabebild. Daneben sind die durch das Strukturtensor-Verfahren ermittelten Salienzkarten für Kurven (b) und Kreuzungspunkte (c) zu sehen. (d) und (e) zeigen die entsprechenden Ergebnisse für das Gabor-Verfahren. Hinweis: Die Ergebnisse sind als Beispiele zu sehen und sollten nicht direkt verglichen werden, weil sich die Parameter der Verfahren grundlegend unterscheiden und nicht einfach aufeinander abgestimmt werden können.

Dieses Kapitel widmet sich der Auswertung des Tensor Voting Verfahrens. Die Umsetzung dieses Themas erweist sich schwieriger als angenommen, denn es gibt eine große Fülle von Gesichtspunkten, die untersucht werden können, sowie eine Reihe jeweiliger Einflussgrößen auf das Verfahren. Dies erschwert die Wahl einer geeigneten Testumgebung, die möglichst konstant gehalten werden sollte, um die unterschiedlichen Ergebnisse besser beurteilen und – soweit möglich – vergleichen zu können. Aus diesem Grund sind die meisten Versuche sehr einfach aufgebaut und verwenden dasselbe Fehlermaß. Außerdem werden häufig Aspekte untersucht, die für die spätere Anwendung in Kapitel 5 relevant sind. Die Auswertung beginnt mit dem „gewöhnlichen“ Tensor Voting Verfahren, das in Kapitel 2 vorgestellt wurde. Im zweiten Teil wird dann die Anwendung auf Grauwertbilder untersucht einschließlich dem Vergleich mit dem Canny-Operator und dem Strukturtensor.

Anmerkung: Aufgrund der großen Anzahl der Eingabe- und Ergebnisbilder in den Versuchen wurden in diesem Kapitel nur einzelne Bilder dargestellt. Der vollständige Satz ist auf der beiliegenden CD zu finden (siehe Anhang C).

4.1 Versuchsaufbau

Die Versuche zur Überprüfung der Qualität und anderer Eigenschaften des Tensor Voting Verfahrens sind wie bereits erwähnt sehr einfach aufgebaut: in der Regel wird das Verfahren auf ein Eingabebild angewendet und das Ergebnis dann mit einer Referenz verglichen, einer sogenannten *Grundwahrheit* (ground truth). Der Vergleich liefert dann ein Fehlermaß zur systematischen Auswertung. Die Ergebnisse sind in Form von Liniendiagrammen und nicht als Tabellen dargestellt, da vielmehr der Verlauf von Werten im Vordergrund steht als deren absoluter Betrag.

4.1.1 Rahmenbedingungen

Damit die Umsetzung der Versuche nicht unnötig verkompliziert wird, müssen mehrere Rahmenbedingungen erfüllt werden:

- Kein Rauschen (außer bei den entsprechenden Tests).
- Zusammenhängende Eingabemerkmale – dadurch wird ein einfacher Vergleich mit dem Originalbild möglich, weil es keine Lücken gibt, die geschlossen werden könnten.

4 Experimente

- Keine illusionären Konturen (siehe Abschnitt 2.5.4) – wiederum wegen des einfachen Vergleichs mit dem Originalbild.
- Einfache Schwarz-Weiß-Bilder – jeder gesetzte Bildpunkt wird als Balltensor interpretiert. Daraus ergeben sich die üblichen zwei Votierdurchgänge: der erste Durchgang, eine spärliche Votierung mit Ballkomponente, dient zur Gewinnung der Richtungsinformation; der zweite Durchgang, eine dichte Votierung nur mit Stabkomponente, dient zur Verfeinerung.

4.1.2 Einflussfaktoren

Durch die Festlegung der Rahmenbedingungen sind die Versuche bereits etwas näher eingegrenzt, jedoch gibt es noch einige Variablen, die sich auf die Ergebnisse auswirken. Sie müssen deshalb konstant gehalten werden, nachdem sie einmal gewählt oder eingestellt worden sind. Es gibt folgende Einflussfaktoren:

- Der Schwellwert T_{cmap} – er bestimmt, welche Pixel in der normalisierten Kurvensaliensz Karte erhalten bleiben und welche als Störung unterdrückt werden. Die Saliensz Karte wird durch Division durch den größten Wert auf das Intervall $[0; 1]$ normalisiert. Soweit nicht anders angegeben, wird der Wert 0.2 verwendet.
- Die Genauigkeit des implementierten Algorithmus – sie spielt insbesondere bei Strukturen eine Rolle, die nicht in Richtung der Hauptachsen orientiert sind, d. h. wenn Rotationen durchgeführt werden müssen (vgl. Gleichung 2.5 und Abschnitt 2.5).
- Die Leistungsfähigkeit des Extraktionsalgorithmus – damit ist gemeint, wie gut Kurven und Kreuzungspunkte aus den entsprechenden Saliensz Karten rekonstruiert werden. Sind die Kurven glatt? Hängen sie zusammen? Wurden die Kreuzungspunkte exakt lokalisiert?

4.1.3 Das Fehlermaß

Für die Beurteilung der Versuchsergebnisse wird die Abweichung der Position einer Struktur von einer Grundwahrheit herangezogen. Die Grundwahrheit stellt das erwartete Ergebnis dar und kann eine analytische Funktion oder auch ein Bild sein. Hier wird nun zuerst der *quadratische Positionsfehler* $\delta_{\text{pos}}^2(p_i)$ ermittelt, indem man die (euklidische) Distanz jedes im Ergebnis gemessenen Punktes p_i zu der tatsächlichen Position τ_i in der Grundwahrheit berechnet und sie anschließend quadriert:

$$\delta_{\text{pos}}^2(p_i) = \|\tau_i - p_i\|_2^2 \quad (4.1)$$

Daraus ermittelt man dann den *mittleren quadratischen Positionsfehler*, der in den Diagrammen angegeben wird:

$$\Delta_{\text{pos}}^2 = \frac{1}{n} \sum_{i=1}^n \delta_{\text{pos}}^2(p_i) \quad (4.2)$$

Die meisten Versuche arbeiten mit einem Kreis (bzw. einer Ellipse) als Grundwahrheit, was die Berechnung des Positionsfehlers sehr vereinfacht. In diesem Fall kennt man den Mittelpunkt m und Radius r des Kreises und berechnet den Fehler wie folgt:

$$\delta_{\text{pos}}^2(p_i) = (\|m - p_i\|_2 - r)^2 \quad (4.3)$$

Anmerkung: In das Fehlermaß gehen in der Regel auch Ausreißer und unerwünscht entstandene, kleine Strukturen ein, sogenannte Artefakte, was den mittleren Fehler deutlich erhöhen kann!

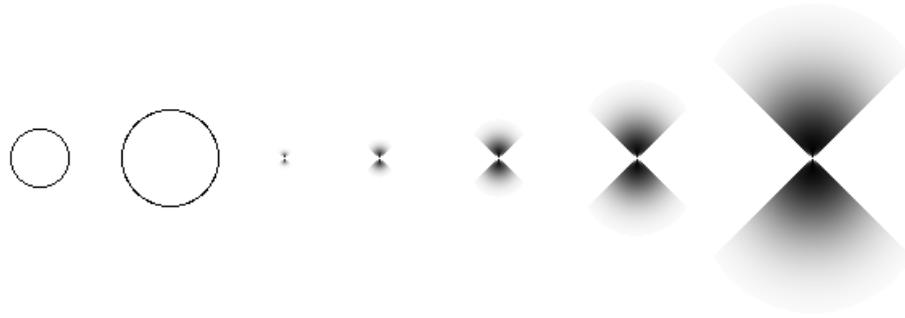


Abbildung 4.1: Größenvergleich unterschiedlicher Testfiguren und Votierfelder. Von links nach rechts sind folgende Figuren dargestellt: zwei Kreise mit Radius 18 bzw. 30 Pixel und fünf Votierfelder mit den Skalenfaktoren $\sigma_v = 3, 6, 12, 24$ und 48.

4.2 Das gewöhnliche Tensor Voting Verfahren

Dieser Abschnitt untersucht einige grundlegende Verhaltensweisen des Tensor Voting Verfahrens. Diese wirken sich entsprechend auch bei den weiter fortgeschrittenen Verfahren aus, die anschließend untersucht werden. Folgende Punkte werden nun in mehreren Versuchen überprüft:

- der Skalenfaktor σ_v der Votierfelder (bezüglich Größe, räumlicher Nähe und Überlagerung von Strukturen),
- Rauschen („Shot Noise“),
- Diskretisierungseffekte bei Rotation von Strukturen und
- Überbrückung von Konkavitäten (als Vorgriff auf die Originalbilder).

4.2.1 Der Skalenfaktor der Votierfelder

Dieser Versuch untergliedert sich in drei Teile, die die Auswirkung des Skalenfaktors (siehe Abschnitt 2.5.4) auf die Größe, die räumliche Nähe und die Überlagerung von Strukturen untersuchen. Der Skalenfaktor wird meistens auch in den anderen Versuchen getrennt berücksichtigt. Zur Veranschaulichung der Größenunterschiede zeigt Abbildung 4.1 die in den Versuchen verwendeten Kreise und Votierfelder.

Größe. Für den ersten Versuch werden Kreise mit einem Radius von 3, 5, 9 und 17 Pixeln verwendet. Wird der Skalenfaktor zu groß gewählt, so entstehen außerhalb der Kreise Artefakte durch Überlagerungseffekte der Votierfelder (gut zu sehen in den Abbildungen 4.2(a) bis (c)). Deswegen wurden zweierlei Fehlerwerte erhoben: Im ersten Fall wurden alle Punkte außerhalb der Kreise ignoriert, d. h. alle Punkte, die 2 Pixel oder weiter außerhalb der Idealkurve liegen, wurden nicht in die Rechnung einbezogen (Abbildung 4.2(e)). Im zweiten Fall wurde der quadratische Positionsfehler für *alle* Punkte berechnet und gemittelt einschließlich der unerwünschten Artefakte im Außenbereich (Abbildung 4.2(f)). Weil sich deren Anzahl mit zunehmender Größe des Skalenfaktors erhöht und diese zusätzlich in immer größeren Abständen von der Idealkurve entstehen, nehmen die Fehlerwerte stetig zu. Bei den gegebenen Kreisen wächst der mittlere quadratische Fehler etwa ab einem Skalenfaktor von 24 auf über 100 an, d. h. durch die Artefakte haben sich die Punkte im Mittel auf 10 Pixel von der Idealkurve entfernt.

Von den Artefakten einmal abgesehen werden die Kreise über einen weiten Bereich des Skalenfaktors hinweg sehr gut lokalisiert; der Fehler beträgt beinahe Null. Zu beachten ist allerdings,

4 Experimente

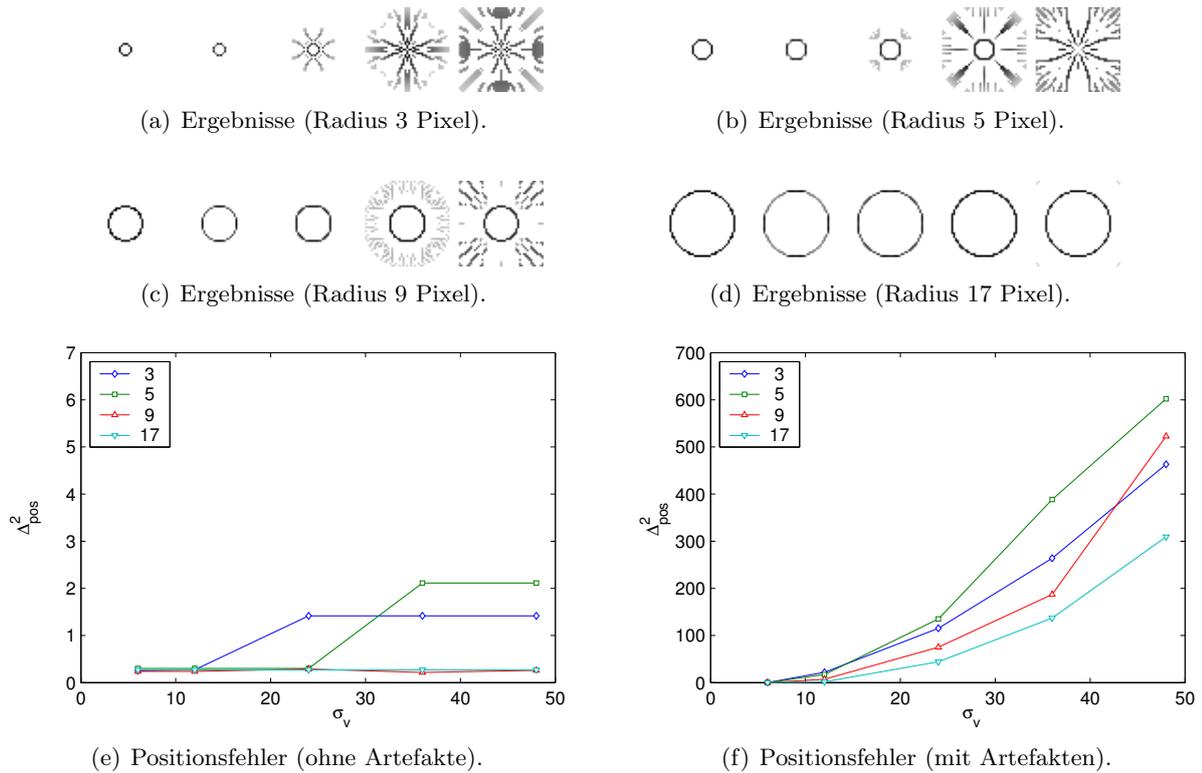


Abbildung 4.2: Einfluss des Skalenfaktors bezüglich Größe einer Struktur. Die Ausschnitte in (a)–(d) zeigen jeweils das Original und die Ergebnisse für $\sigma_v = 6, 12, 24$ und 48 . In (d) sind die Artefakte im Außenbereich abgeschnitten und nicht mehr zu sehen. (e) und (f) zeigen die Positionsfehler für unterschiedliche Radien, einmal ohne und einmal mit Berücksichtigung der Artefakte im Außenbereich. Hinweis: Die Diagramme haben eine unterschiedliche Skala!

dass bei einem sehr großen Skalenfaktor kleine Strukturen aufgebrochen werden und verloren gehen (deshalb der Anstieg der Fehlerwerte in Abbildung 4.2(e)).

Räumliche Nähe. Dieser und der nächste Versuch dienen mehr zur Veranschaulichung als zur Auswertung und es wurde kein Fehlermaß bestimmt. Für den zweiten Versuch werden eine Linie und ein Kreisstück (mit einem Radius von 18 Pixeln) in verschiedenen großen Abständen voneinander platziert (zu sehen in Abbildung 4.3). Es wurden keine symmetrischen Figuren gewählt, z. B. zwei Halbkreise, weil damit fast keine sichtbaren Effekte auftreten.

Die Bildern zeigen, dass die Tensorvotierung sehr gut mit nahe beieinander liegenden Strukturen zurechtkommt und erst bei entsprechend großen Skalen beginnen die Strukturen, sich gegenseitig so zu beeinflussen, dass sie ihre ursprüngliche Form verändern, hier wird beispielsweise die Linie an den Enden leicht verformt, sozusagen „abgestoßen“, und bei geringer Distanz entsteht eine Verbindungskurve zwischen den Linien- und Kreisenden.

Überlagerung. Für den dritten Versuch werden zwei Kreisausschnitte (wiederum mit einem Radius von 18 Pixeln) in unterschiedlich großem Abstand überlagert. Es zeigt sich, dass das Verfahren bereits ab einer Lücke von einem Pixel (d. h. zwei Pixel Entfernung) die Kreise unterscheiden und sehr gut extrahieren kann, unabhängig von der Distanz (siehe Abbildung 4.4). Wie bereits im letzten Versuch festgestellt muss die Skala sehr groß werden, damit die Strukturen miteinander verschmelzen.

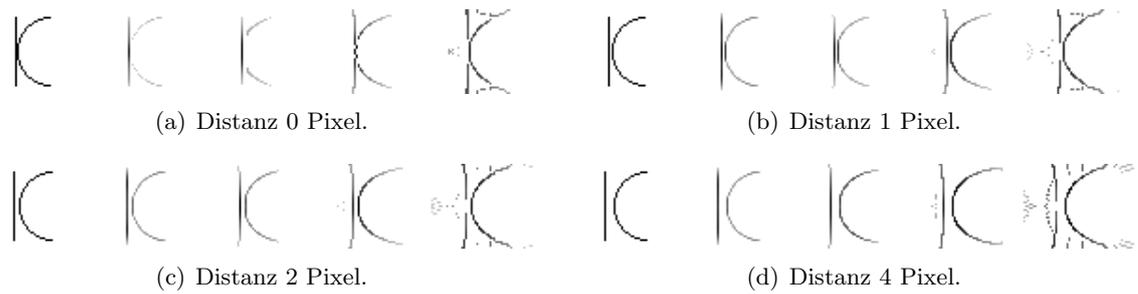


Abbildung 4.3: Einfluss des Skalenfaktors bezüglich Distanz zweier Strukturen (die Ausschnitte zeigen jeweils das Original und die Ergebnisse für $\sigma_v = 6, 12, 24$ und 48). Die Halbkreise haben einen Radius von 18 Pixeln.

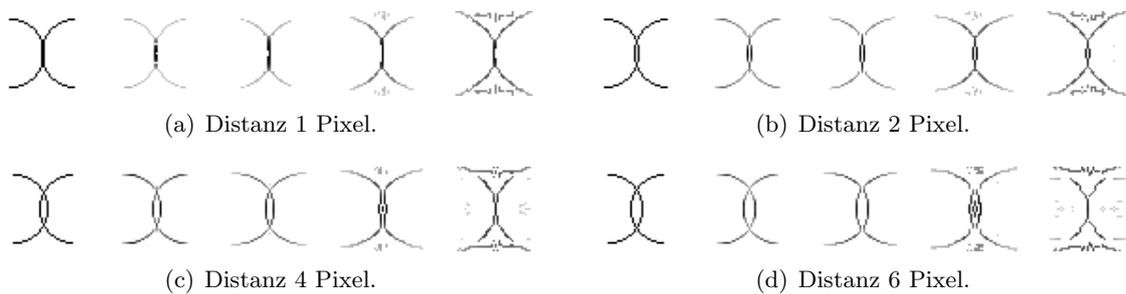


Abbildung 4.4: Einfluss des Skalenfaktors bezüglich Überlagerung zweier Strukturen (die Ausschnitte zeigen jeweils das Original und die Ergebnisse für $\sigma_v = 6, 12, 24$ und 48). Die Halbkreise haben einen Radius von 18 Pixeln.

4.2.2 Rauschen

Es gibt unterschiedliche Arten von Rauschen in einem Bild mit verschiedenen Ursachen. Beispiele dafür sind Gauß'sches Rauschen und „Shot Noise“ (auch „Salt-and-Pepper Noise“ genannt), die durch zufällige Fehler bei der Messung bzw. Aufnahme von Bildern entstehen können. Gauß'sches Rauschen kann künstlich erzeugt werden, indem jeder Bildpunkt mit einem standardnormalverteilten zufälligen Wert $Z_n \in \mathbb{R}$ überlagert wird, so dass beliebige Grauwerte entstehen können:

$$g(x, y) \mapsto g(x, y) + \sigma Z_n + \mu \quad (4.4)$$

wobei σ die Standardabweichung und μ den Mittelwert des Rauschens angibt. σ wird in den Versuchen mit Werten zwischen 0 und 0.4 und μ mit 0 belegt. Werte, die außerhalb der zulässigen Intensitätswerte (zwischen 0 und 1) liegen, werden auf den kleinst- bzw. größtmöglichen Wert gesetzt.

▷ **Beispiel 4.1** Überlagert man einen Bildpunkt, der den Intensitätswert $i = 0.5$ hat, mit Gauß'schem Rauschen mit der Standardabweichung $\sigma = 0.2$ und dem Mittelwert $\mu = 0$, so liegt das Ergebnis aufgrund der Normalverteilung zu 68 % zwischen $i - \sigma$ und $i + \sigma$, also zwischen 0.3 und 0.7 (siehe z. B. [Fis89]).

Bei „Shot Noise“ besteht an jedem Bildpunkt eine gewisse Chance, dass der Wert gestört ist und damit entweder auf den kleinsten oder größten Intensitätswert gesetzt wird, d. h. auf Schwarz oder Weiß. Man erzeugt es mit Hilfe eines gleichverteilten zufälligen Wertes $Z_g \in [0; 1]$ und dem

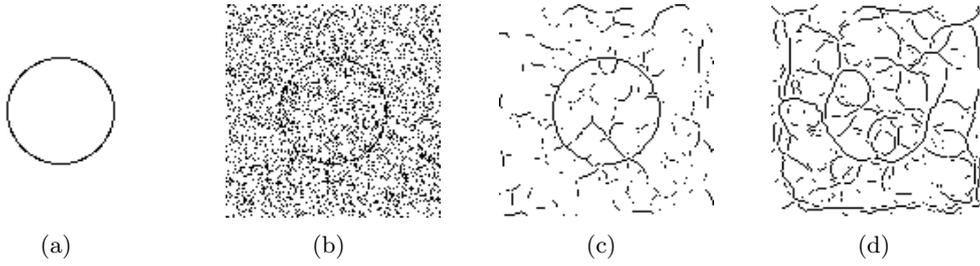


Abbildung 4.5: Ein Beispiel für „Shot Noise“: (a) Binäres Originalbild (der Kreisradius beträgt 30 Pixel, die Bildgröße 121×121 Pixel), (b) Original mit 40 % „Shot Noise“, (c) extrahierte Strukturen für $\sigma_v = 6$. (d) Ein Beispiel für die Entstehung „willkürlicher“ Strukturen bei starkem Rauschen (50 %) und einem großen Skalenfaktor $\sigma_v = 12$.

Anteil p des Rauschens:

$$g(x, y) \mapsto \begin{cases} 0 & Z_g < p/2 \\ 1 & Z_g \geq p/2 \text{ und } Z_g < p \\ g(x, y) & Z_g \geq p \end{cases} \quad (4.5)$$

Da die Betrachtung von Gauß'schem Rauschen bei den für das normale Verfahren verwendeten Schwarz-Weiß-Bildern keinen Sinn hat, wurde an dieser Stelle nur das „Shot Noise“ untersucht. Wegen der zufälligen Verteilung der Störungen muss dieser Versuch mehrmals wiederholt werden (insgesamt N -mal), um repräsentative Ergebnisse zu erhalten.

Abbildung 4.5 zeigt das binäre Originalbild, das für diesen Versuch verwendet wird, und die extrahierten Strukturen bei einem stark verrauschten Bild. Abbildung 4.6 zeigt den Verlauf der Fehlerwerte für unterschiedlich große Skalen und unterschiedlich stark verrauschte Bilddaten. In die Fehlerwerte gehen alle Punkte im Ergebnis ein, insbesondere auch die willkürlich durch das Rauschen entstandenen Artefakte. Es wird wie üblich der quadratische Abstand zum Idealkreis berechnet und die Einzelwerte anschließend gemittelt. Bis zu einem gewissen Anteil an Rauschen (hier etwa $p = 0.3$) verbessern sich die Ergebnisse mit Vergrößerung des Skalenfaktors (durch die glättende Wirkung). Dies ist z. B. bei $p = 0.2$ sehr deutlich zu sehen: hier ist der Fehler bei einem Skalenfaktor von $\sigma_v = 12$ zwei- bzw. fünffmal kleiner als bei einem Faktor von $\sigma_v = 6$ bzw. $\sigma_v = 3$; der Kreis wird gut detektiert und es entstehen nur wenige Artefakte um die

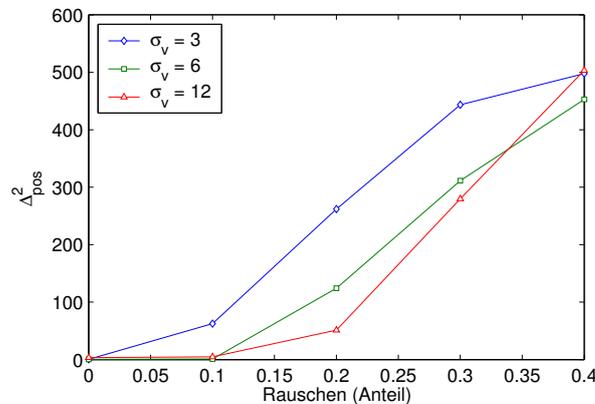


Abbildung 4.6: Einfluss von „Shot Noise“ auf das Tensor Voting Verfahren ($N = 20$, $T_{\text{cmap}} = 0.2$). Dargestellt sind die Positionsfehler für unterschiedliche Skalenfaktoren σ_v . Das Rauschen ist anteilmäßig angegeben (0.4 entsprechen 40 %).

Kreislinie herum. Nimmt das Rauschen weiter zu ($p > 0.2$), so weicht die extrahierte Kurve an manchen Stellen von der Kreislinie ab und es entstehen auch in den Außenbereichen des Bildes Artefakte. Bei noch stärkerem Rauschen (ab etwa 40% Rauschen) kann dann selbst ein größerer Skalenfaktor keine sinnvollen Ergebnisse mehr erzielen: die Kreislinie bricht auf und es entstehen „willkürliche“ Strukturen (siehe Abbildung 4.5(d)). Die Ergebnisbilder zeigen außerdem, dass bei größeren Skalen längere, zusammenhängende Artefakte entstehen, was auch zu erwarten ist, denn durch die größeren Votierfelder können Strukturen (auch unerwünschte) besser gruppiert werden.

4.2.3 Diskretisierungseffekte bei Rotation von Strukturen

Um den Einfluss der Orientierung einer Struktur auf das Ergebnis der Tensorvotierung zu untersuchen, wurden zwei orthogonale Linien Schritt für Schritt gedreht und jeweils die Anzahl der sich unterscheidenden Bildpunkte vom Eingabe- zum Ergebnisbild ermittelt. Die Linien wurden in 3° -Schritten gedreht, wobei jeweils etwa die Hälfte aller Pixel ihre Position ändern. Die Ergebnisse zeigen – wenn überhaupt – nur vereinzelte Unterschiede auf (in der Testreihe mit 15 Orientierungen nur bei drei Bildern, ein Beispiel dafür ist in Abbildung 4.7 zu sehen). Die Differenzen entstehen durch den Non-Maximum Suppression Algorithmus, der die Linien aus der Kurvensalienzkarte extrahiert. Sind die lokalen Maxima in der Salienzkarte nicht eindeutig, so kann es passieren, dass der Algorithmus andere Pixel als im Original auswählt. Wenn beispielsweise die Mitte eines acht Pixel breiten Bandes bestimmt werden muss, kann der Algorithmus entweder das vierte oder fünfte Pixel auswählen. Eine Glättung der Eingabe vor der Tensorvotierung (\Rightarrow Anti-Aliasing) ändert nichts an den Ergebnissen, weil die Tensorvotierung selbst eine Art Glättung durchführt und die Unterschiede erst *nach* der Votierung bei der Extraktion der Linien entstehen. Abgesehen von diesem hier beschriebenen Effekt hat die Orientierung einer Struktur keinen Einfluss auf das Ergebnis der Tensorvotierung.

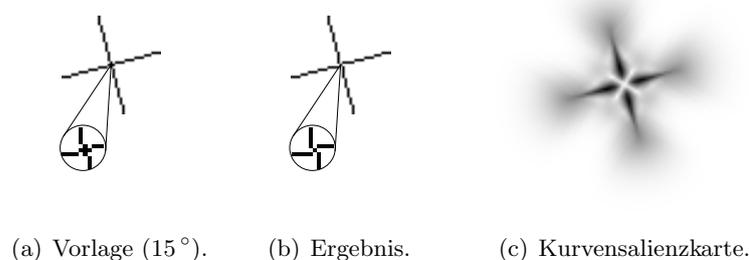


Abbildung 4.7: Einfluss der Orientierung zweier orthogonaler Linien auf die Qualität des Ergebnisses aufgrund von Diskretisierungseffekten. (a), (b) Bei einer Drehung von 15° entstehen die Pixel-Unterschiede im Zentrum. Das Ergebnis in (b) wurde aus der Kurvensalienzkarte in (c) ermittelt.

4.2.4 Überbrückung von Konkavitäten

In diesem Versuch wird die innere Kontur der linken Herzkammer durch eine große Ellipse mit zwei Konkavitäten nachgebildet (siehe Abbildung 4.8(a) und (b), vgl. Abbildung 5.2). Die Konkavitäten stellen dabei die Papillarmuskeln dar, die ausgespart und durch einen glatten Bogen ersetzt werden sollen, so dass wieder eine vollständige Ellipse entsteht (bzw. übertragen auf die Originalbilder eine glatte Endokontur). Zwei Parameter sind hierbei von Interesse: zum einen der Radius p der Konkavitäten und zum anderen der Skalenfaktor σ_v , der Votierfelder. Auf diese Weise soll untersucht werden, welche Größe die Konkavitäten haben dürfen, damit sie noch

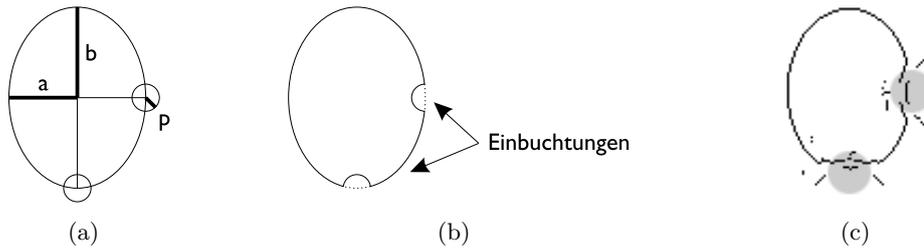


Abbildung 4.8: Konstruktion der Testfigur mit Konkavitäten. (a) Von einer Ellipse mit den Radien a und b werden zwei Kreise mit Radius p ausgespart, deren Zentren am unteren bzw. am rechten Schnittpunkt der Ellipse mit ihren Hauptachsen liegen. (b) Die entstandene Figur mit den Aussparungen und der gewünschten Fortsetzung (gestrichelt). (c) Das Ergebnis der Tensorvotierung für $p = 7$ und $\sigma_v = 24$; grau hinterlegt sind die Bereiche, die bei spezieller Betrachtung der Konkavitäten ausgewertet werden.

mittels der Tensorvotierung überbrückt werden können. Die Größe der Ellipse (angegeben durch die zwei Radien $a = 30$ und $b = 40$) beeinflusst ab einem bestimmten Wert die Ergebnisse nur noch geringfügig, weshalb sie den Originalbildern nachempfunden und in den Versuchen konstant gehalten wird. Werden die Radien zu klein gewählt, z. B. $a = 15$ und $b = 20$, so liegen die Konkavitäten zu nahe beieinander und können sich überschneiden, so dass die Figur nicht mehr die geforderte Gestalt hat. Anders als in den Originalbildern wird hier jedoch die Exzentrizität der Ellipse erhöht, so dass sich die Krümmung an den beiden Konkavitäten stärker unterscheidet.

Es werden zwei Testläufe durchgeführt, wobei beim ersten Durchlauf speziell nur die Bereiche der Konkavitäten (siehe Abbildung 4.8(c)) und beim zweiten das gesamte Bild einschließlich der entstehenden Artefakte berücksichtigt werden. Bei der Betrachtung der Konkavitäten werden diese zwei Bereiche mit jeweils einem ausgefüllten Kreis maskiert, dessen Radius um ein Pixel größer ist als der Radius der Konkavitäten. Nur Punkte innerhalb dieser Masken gehen dann in die Fehlerberechnung ein. Abbildung 4.9(a) zeigt die Fehlerwerte bei der Betrachtung der Konkavitäten. Man sieht, dass sie sich mit wachsender Größe der Votierfelder (zunehmendes σ_v) verringern. Dieser Sachverhalt ist einfach nachzuvollziehen: bei einem größeren Votierfeld übt auch eine größere Nachbarschaft einen Einfluss auf eine Stelle aus und der längere Ellipsenbogen kann sich gegenüber den kurzen Segmenten der Konkavitäten durchsetzen. Allerdings zeigen die Ergebnisbilder, dass bei einem größeren Skalenfaktor die durchgehende Kurve unterbrochen wird und ein versetztes Segment entsteht, das von zwei nach innen ziehenden Kurvenstücken flankiert wird (siehe Abbildung 4.8(c)). Vollständig ausgeglichen werden nur die Konkavitäten mit 3 Pixel Radius (der mittlere quadratische Fehler geht gegen Null), bei einem Radius von 5 Pixel sinkt die mittlere Abweichung vom Ellipsenbogen immerhin ab einem Skalenfaktor von etwa 15 auf unter 3 Pixel und die Konkavitäten werden noch gut überbrückt. Bei größeren Konkavitäten dagegen kann nur noch ein sehr großer Skalenfaktor ($\sigma_v > 30$) einen Ausgleich schaffen, wobei jedoch die oben genannten Unterbrechungen entstehen.

Die Fehlerwerte in Abbildung 4.9(b) bei Betrachtung des *gesamten* Bildes zeigen einen anderen Trend. Wie bereits im Versuch zur Größe des Skalenfaktors festgestellt (siehe Abschnitt 4.2.1) entstehen bei großen Votierfeldern Artefakte. Dies geschieht in diesem Versuch genauso, anfangs nur in der Nähe der Konkavitäten, aber mit wachsendem Skalenfaktor auch auf beiden Seiten entlang des Ellipsenbogens. Die andere Beobachtung ist, dass die „spitzen Ecken“ der Konkavitäten geglättet werden und nach innen wandern und nicht dass das Kurvensegment der Konkavität wie vorgesehen nach außen wandert. Diese beiden Phänomene führen zu einem Anstieg des mittleren quadratischen Fehlers ab einem Skalenfaktor von $\sigma_v = 15$. Die besseren lokalen Ergebnisse gehen also mit einer Verschlechterung der globalen Ergebnisse einher. Das Problem bei der Über-

Überbrückung der Konkavitäten besteht darin, dass deren Kurvensegmente hohe Salienzen erzeugen, die die Salienzen an den Punkten der geraden Fortführung klein erscheinen lassen.

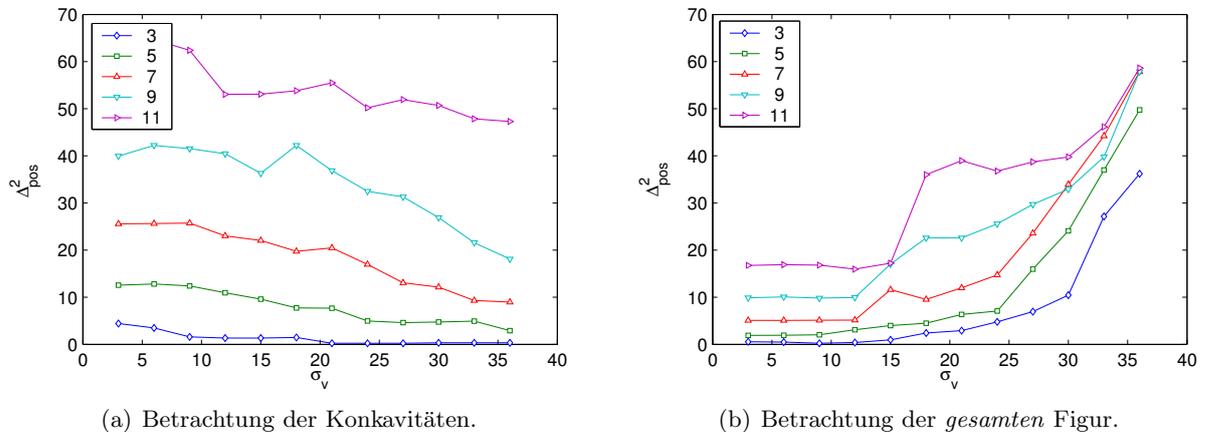


Abbildung 4.9: Ergebnisse für die Überbrückung von Konkavitäten. Jede Linie stellt die mittlere quadratische Abweichung der betrachteten Punkte von der Idealposition für einen bestimmten Radius p und unterschiedliche Skalenfaktoren σ_v dar. Hinweis: Die Kurvensegmente außerhalb der Konkavitäten werden korrekt erkannt und senken deshalb bei der Betrachtung der gesamten Figur den mittleren quadratischen Positionsfehler (im Vergleich von (a) zu (b)). Für einen Größenvergleich der Figur mit den Votierfeldern kann der zweite Kreis in Abbildung 4.1 herangezogen werden.

4.3 Tensor Voting gegenüber anderen Verfahren

Hinter diesem Versuch steht die Frage, welche Vorteile die Tensorvotierung gegenüber einfacheren Verfahren bringt. Außerdem werden zum ersten Mal Versuche auf Grauwertbildern durchgeführt, so dass dieser Abschnitt auch als Auswertung der Tensorvotierung auf Grauwertbildern gesehen werden kann. Dabei ist zu beachten, dass die Testfigur in der Vorlage ausgefüllt sein muss, denn eine Linie würde aufgrund der in den Verfahren verwendeten Gradientenberechnung zwei Ausschläge liefern (vgl. Abschnitt 3.2.3). Es werden die folgenden drei Verfahren anhand einiger einfacher Kriterien verglichen:

- *Canny-Operator*. Dieses Verfahren wurde in den Versuch aufgenommen, um einen Vergleich mit einem „klassischen“, aber dennoch sehr leistungsfähigen Ansatz der Kontrastdetektion zu haben (denkbar wäre sicherlich auch ein Vergleich mit einem rudimentären Verfahren wie z. B. dem Sobel-Operator). Wie bereits in Abschnitt 3.2.1 beschrieben eignet es sich sehr gut zur Kantendetektion und wird durch drei Parameter charakterisiert: die Standardabweichung σ_g des Gauß'schen Tiefpassfilters und die zwei Schwellwerte T_u und T_o des adaptiven Schwellwertverfahrens. Für σ_g werden unterschiedliche Werte gewählt und die Schwellwerte je nach Eingabebild automatisch anhand des Histogramms bestimmt. Als oberer Schwellwert T_o wird derjenige Intensitätswert gewählt, der größer oder gleich etwa 70% aller Intensitätswerte des Bildes ist. Der untere Schwellwert T_u ist gleich $0.4 \cdot T_o$. Diese automatische Wahl der Schwellwerte stellt eine praktische Vereinfachung dar und erlaubt eine breite Anwendung des Operators. Ihr Einfluss wird in den einzelnen Versuchen erläutert.
- *Strukturtensor*. Der Strukturtensor wird als Vorverarbeitungsstufe für die spätere Tensorvotierung verwendet, hier jedoch selbständig ohne anschließende Weiterverarbeitung. Damit

soll im Vergleich der Beitrag des Tensor Voting Verfahrens erkennbar werden. Der Strukturtensor wird durch die Methode der Gradientenberechnung (hier mittels Sobel-Operator) und den Glättungsfilter (hier ein Gauß'scher Tiefpassfilter mit Standardabweichung σ_g) beschrieben. Durch das Verfahren wird eine Tensorkarte und daraus eine Kurvensalienzkarte berechnet (siehe Abschnitt 3.3.1). Aus letzterer werden identisch zum Tensor Voting Verfahren mit Hilfe des Schwellwertes T_{cmap} und einer Non-Maximum Suppression Methode die Kurven extrahiert (siehe Abschnitt 2.6.1). Der Schwellwert wird wie in den anderen Versuchen mit 0.2 belegt und ist weniger entscheidend für die Qualität der Ergebnisse.

- *Strukturtensor + Tensorvotierung.* Schließlich kommt das Tensor Voting Verfahren zum Einsatz, das den Strukturtensor zur Vorverarbeitung verwendet. Es wird nur durch den Skalenfaktor σ_v der Votierfelder charakterisiert. Die Kurven werden wie im letzten Abschnitt beschrieben aus der Kurvensalienzkarte extrahiert.

Folgende Kriterien werden nun in mehreren Versuchen überprüft. Sie wurden gewählt, weil sie auch in den Originaldaten eine wichtige Bedeutung haben (vgl. Abbildung 5.3):

- rampenförmige Kanten (verhältnismäßig breite Kanten mit einem rampenförmigen Grauwertverlauf),
- schwache Kontraste und
- Rauschen („Shot Noise“ sowie Gauß'sches Rauschen).

4.3.1 Rampenförmige Kanten

Um einen „weichen“ Kontrast zu erzeugen, wird eine Stufenkante mit einem Gauß-Filter mit der Standardabweichung σ_{kante} (oder einem anderen Glättungsfilter) geglättet. Auf diese Weise entsteht an der Kante ein rampenförmiger Grauwertverlauf (eine sogenannte „ramp edge“). Als Figur wird ein Kreis mit einem Radius von 30 Pixeln verwendet (zweite Figur von links in Abbildung 4.1). Wegen der Artefakte, die durch die Tensorvotierung entstehen, werden in diesem Versuch wiederum zweierlei Fehlerwerte ermittelt. Zum einen wird nur eine schmale Nachbarschaft um die Idealkurve herum betrachtet (Abbildung 4.10(a)) und zum anderen das gesamte Bild einschließlich aller Artefakte (Abbildung 4.10(b)). Die Artefakte entstehen in diesem Versuch hauptsächlich im Inneren des Kreises – weitaus weniger außerhalb des Kreises – und haben einigen Abstand von der Kreislinie. Deshalb werden hier bei der Fehlerwertberechnung alle Punkte ignoriert, die innerhalb 85 % und außerhalb 125 % des Kreisradius liegen. Diese Maske genügt als einfache Heuristik, um die meisten Störungen auszuschließen. Die Wahl der Schwellwerte T_o und T_u beim Canny-Operator wirken sich in diesem Versuch nicht aus.

Bei allen drei Verfahren wird die Kante mit einer gewissen Abweichung stets in der „Mitte“ des Grauwertverlaufs lokalisiert. Je nach Breite der rampenförmigen Kante im Original entfernt sich die gefundene Kontur oder nähert sich wieder an. In Abbildung 4.10(a) spiegelt sich die Abweichung in dem beinahe periodischen Verlauf der Fehlerkurven wider. Sie steigt auf drei bis vier Pixel Entfernung an und nimmt dann wieder ab. Für die Tensorvotierung ist ein solches Beispiel in Abbildung 4.10(c) zu sehen. Die Maxima unterscheiden sich bei den drei Verfahren und liegen beim Canny-Operator bei $\sigma_{\text{kante}} = 16$, beim Strukturtensor und bei der Tensorvotierung bei $\sigma_{\text{kante}} = 10$ und $\sigma_{\text{kante}} = 24$. Die Kurve der Tensorvotierung mit einem großen Skalenfaktor von $\sigma_v = 12$ weicht allerdings ab: die höheren Fehlerwerte bei geringer Glättung stammen von Artefakten, die dicht außerhalb der Kreislinie entstehen; außerdem ist die maximale Abweichung hin zu $\sigma_{\text{kante}} = 18$ verschoben.

In Abbildung 4.10(b) ist bei der Tensorvotierung deutlich der Einfluss der Artefakte zu sehen. Anders als in den bisherigen Versuchen unterscheidet sich die Anzahl und Position der Störungen

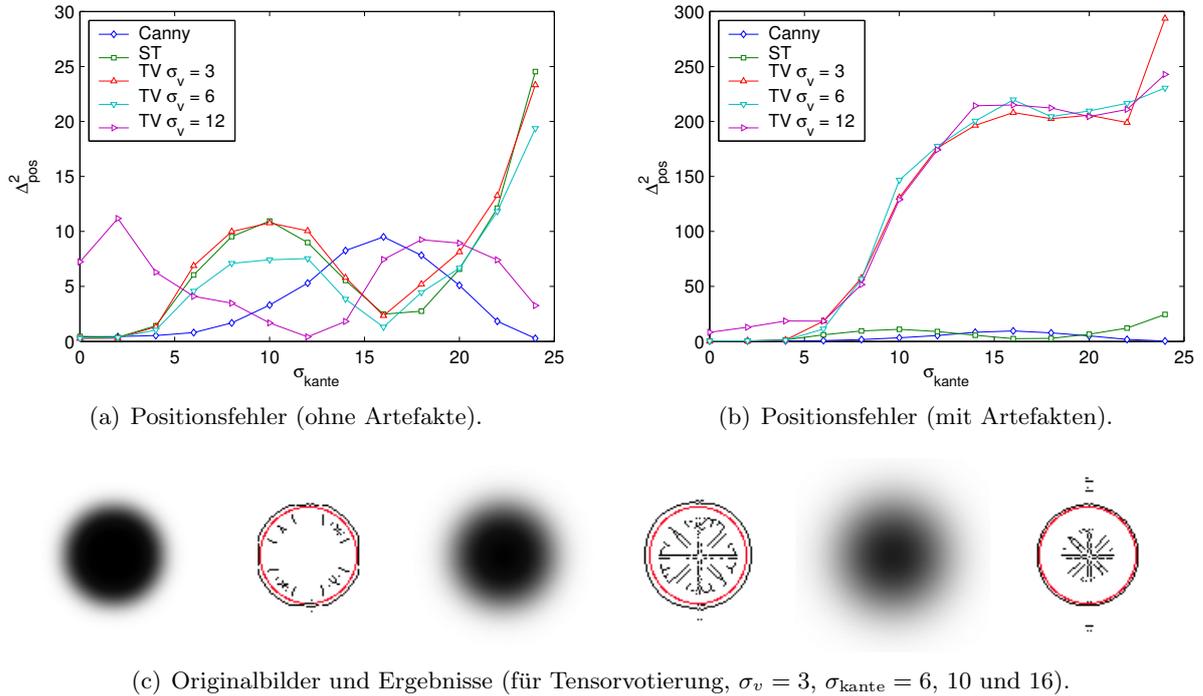


Abbildung 4.10: Ergebnisse unterschiedlicher Verfahren bei rampenförmigen Kanten ($\sigma_g = 1, T_{\text{cmap}} = 0.2$). Die Linien stehen für den Canny-Operator (Canny), den Strukturtenor (ST) und die Tensorvotierung (TV) mit unterschiedlichen Skalenfaktoren σ_v . In (c) sind die Originalkreise zum Vergleich rot eingezeichnet. Hinweis: Die Diagramme haben eine unterschiedliche Skala!

kaum bei den unterschiedlichen Skalenfaktoren. Abgesehen von den erhöhten Fehlerwerten bei einem Skalenfaktor von $\sigma_v = 12$ verlaufen die Fehlerkurven nahezu identisch, d. h. die Entstehung der Artefakte wird in diesem Fall viel stärker durch die Bildstruktur als durch die Größe des Skalenfaktors begünstigt. Alle drei Verfahren schneiden in diesem Versuch vergleichbar ab und lokalisieren die Kante entsprechend gut. Allerdings fällt die Entstehung der Artefakte bei der Tensorvotierung negativ auf.

4.3.2 Schwache Kontraste

Für diesen Versuch wird ein halbiertes Kreis mit einem Radius von 30 Pixeln, dessen eine Hälfte schwarz, die andere grau ist, als Testfigur herangezogen. Die schwarze Fläche dient dabei gewissermaßen als Referenz und sollte in jedem Fall erkannt werden. Die Intensitätswerte des Bildes sind auf das Intervall $[0; 1]$ begrenzt. Die Intensität der grauen Fläche wird mit ΔI als Differenz zum Hintergrund angegeben. Um den möglichen Einfluss der Intensitätsdifferenz besser untersuchen zu können, werden deren unterschiedlichen Werte nicht in einem linearen, sondern in einem logarithmischen Abstand ausgewertet. Dazu wird außerdem anstelle des üblichen Schwellwertes $T_{\text{cmap}} = 0.2$ ein niedrigerer Wert von 0.01 verwendet.

In Abbildung 4.11(b) sind drei Beispiele für Eingabe- und Ergebnisbilder zu sehen. Wie erwartet wird der schwarz gefüllte Halbkreis in jedem Fall erkannt, die obere Hälfte mit schwachem Kontrast nur bis zu einem bestimmten Helligkeitswert. Abbildung 4.11(a) zeigt das Diagramm mit den ermittelten Fehlerwerten. Abgesehen von einigen geringen Abweichungen entspricht deren Verlauf einer Stufenfunktion, jedoch fallen die Werte bei der Tensorvotierung mit $\sigma_v > 3$ nicht auf Null. Betrachtet man dazu die Ergebnisbilder, so erkennt man erneut einige Artefakte, die wie eine Korona die extrahierten Kurven umgeben. Das eigentlich interessante Ergebnis ist

4 Experimente

der Sprung der Fehlerwerte bei $\Delta I = 2 \cdot 10^{-2}$ (beim Canny-Operator) bzw. bei $\Delta I = 1.4 \cdot 10^{-1}$ (bei den anderen Verfahren). Er markiert den Punkt, ab dem der schwache Kontrast der oberen Kreishälfte erkannt wird, und hängt direkt von den Schwellwerten der Verfahren ab. Der Canny-Operator schneidet in diesem Versuch etwas besser als die anderen Verfahren ab, jedoch liegt das lediglich an der Wahl der Schwellwerte. Die automatische Anpassung der Schwellwerte beim Canny-Operator (siehe Abschnitt 4.3) hilft bis zu einem gewissen Grad, mit den Gegebenheiten des Bildes besser zurecht zu kommen. Durch eine Anpassung des Schwellwertes T_{cmap} kann man allerdings mit den anderen Verfahren dieselben Ergebnisse erzielen.

Die Versuche haben gezeigt, dass beliebig schwache Kontraste erkannt werden können (im Rahmen der Rechengenauigkeit), sofern die Schwellwerte für die Merkmalsextraktion entsprechend angepasst werden. Bei echten Bildern darf diese untere Schwelle jedoch nicht zu niedrig gewählt werden, weil sonst neben den gesuchten Konturen zu viele Artefakte entstehen. Wiederum entstehen bei der Tensorvotierung mit großen Skalenfaktoren kleine Störungen.

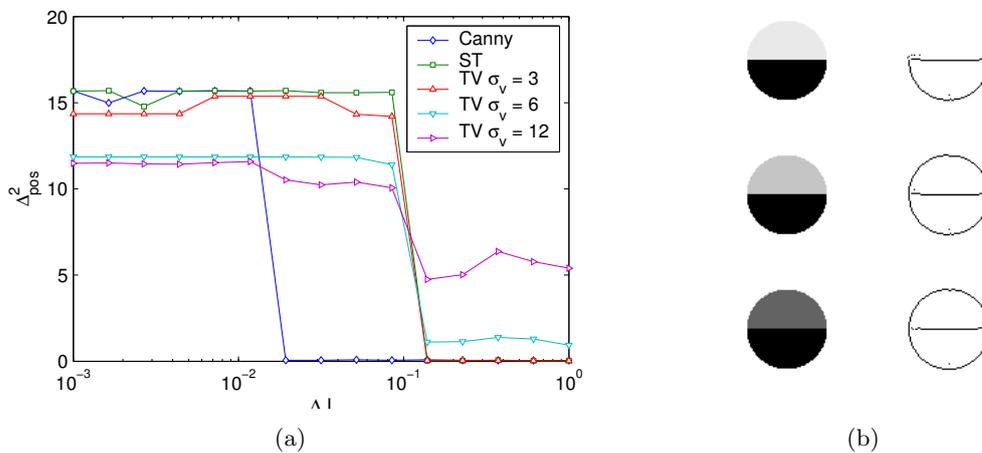


Abbildung 4.11: Ergebnisse unterschiedlicher Verfahren bei schwachen Kontrasten. (a) Das Liniendiagramm mit den Positionsfehlern ($\sigma_g = 1$, $T_{\text{cmap}} = 0.01$). (b) Originalbilder und Ergebnisse der Tensorvotierung ($\sigma_v = 3$, $\Delta I = 0.08, 0.22$ und 0.61). Die Linien stehen für den Canny-Operator (Canny), den Strukturtenor (ST) und die Tensorvotierung (TV) mit unterschiedlichen Skalenfaktoren σ_v .

4.3.3 Rauschen

Dieser Versuch wertet entsprechend dem Versuch in Abschnitt 4.2.2 einen Kreis mit einem Radius von 30 Pixeln aus, der mit Rauschen überlagert worden ist. Da an dieser Stelle jedoch mit Grauwertbildern gearbeitet wird, kommt neben dem „Shot Noise“ auch Gauß'sches Rauschen zum Einsatz. Neben den zwei Arten von Rauschen werden die Versuche für unterschiedliche Standardabweichungen σ_g des Gauß'schen Tiefpassfilters durchgeführt, der sowohl vom Canny-Operator als auch vom Strukturtenor-Verfahren verwendet wird.

Die Ergebnisse der Versuche sind in Abbildung 4.12 zu sehen. Sie fallen je nach Art des Rauschens und Stärke der Gauß'schen Glättung unterschiedlich aus. Der Canny-Operator liefert ähnliche Ergebnisse unabhängig von der Art des Rauschens. Bei einem kleinen Glättungsfaktor ($\sigma_g = 1$) kann er das Rauschen nicht unterdrücken, was an den Fehlerwerten zu sehen ist, die konstant bei etwa 600 liegen und sich bei zunehmendem Rauschen nicht mehr ändern (Abbildung 4.12(a) und (c)). Ein größerer Glättungsfaktor ($\sigma_g = 4$) wirkt sich dagegen positiv aus, d. h. schwaches „Shot Noise“ oder Gauß'sches Rauschen wird sehr erfolgreich unterdrückt (Anteil bzw.

Standardabweichung 0.1, Abbildung 4.12(b) und (d)). Es treten lediglich vereinzelt Störungen auf (kurze Kurvenstücke), der Kreis selbst wird jedoch gut detektiert. Bei zunehmendem Rauschen entstehen außerhalb des Kreises immer mehr Störungen, die durch die relativ große Glättung lange Ketten bilden können. Stellenweise weicht der extrahierte Kreis leicht von der Idealform ab.

Bei den übrigen vier Versuchsreihen fallen die Ergebnisse etwas anders aus. Das Strukturtensor-Verfahren kommt mit Gauß'schem Rauschen eindeutig besser zurecht als mit „Shot Noise“. Bei Gauß'schem Rauschen liefert es bei beiden Glättungsfaktoren dieselben hochwertigen Ergebnisse (Abbildung 4.12(c) und (d)). Die Fehlerwerte liegen praktisch bei Null und erst bei starkem Rauschen (Standardabweichung 0.4) entstehen einige, kleine Störungen im gesamten Bildbereich. Mit „Shot Noise“ kann der Strukturtensor etwas schlechter umgehen (Abbildung 4.12(a) und (b)). Artefakte entstehen bereits bei leichtem Rauschen (Anteil 0.2) und die extrahierte Kreislinie ist unregelmäßiger. Ein größerer Glättungsfaktor verschlechtert hier die Ergebnisse. Zwar entstehen weniger Artefakte, aber auch der Kreis wird schlechter detektiert, deutlich zu sehen in dem Anstieg der Fehlerwerte bei einem Rauschanteil von 0.4 (hier werden nur noch wenige Teile des Kreises extrahiert).

Die Ergebnisse der Tensorvotierung orientieren sich an denen des „vorgeschalteten“ Strukturtensor-Verfahrens. Die Tensorvotierung bringt gegenüber dem reinen Strukturtensor nur bei „Shot

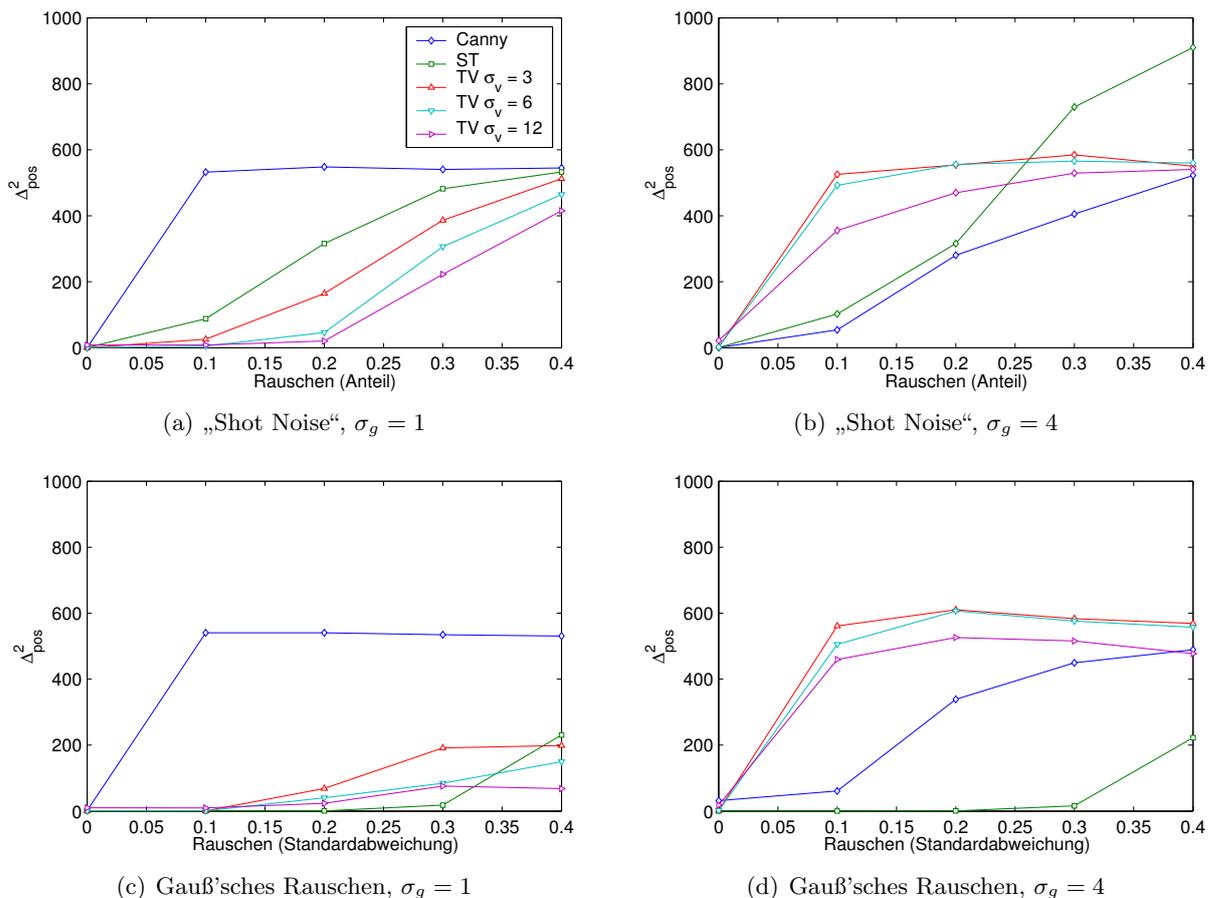


Abbildung 4.12: Einfluss von Rauschen auf unterschiedliche Verfahren ($T_{\text{cmap}} = 0.2$, die Werte wurden wegen der Zufälligkeit des Rauschens aus 20 Wiederholungen gemittelt). Die Linien stehen für den Canny-Operator (Canny), den Strukturtensor (ST) und die Tensorvotierung (TV) mit unterschiedlichen Skalierungsfaktoren σ_v .

4 Experimente

Noise“ und starkem Gauß'schem Rauschen einen Vorteil (jeweils mit $\sigma_g = 1$). Die Fehlerwerte sinken leicht mit zunehmendem Skalenfaktor σ_v und die extrahierte Kreislinie ist regelmäßiger. Es stellt sich insbesondere heraus, dass ein kleiner Glättungsfaktor bei der Vorverarbeitung (mit dem Strukturtenor-Verfahren) bessere Ergebnisse liefert als ein großer ($\sigma_g = 4$). Bei einem größeren Glättungsfaktor findet das Tensor Voting Verfahren in dem Rauschen viel eher zusammenhängende Strukturen, die jedoch willkürlich und unerwünscht sind (zu sehen in den früh ansteigenden, hohen Fehlerkurven in Abbildung 4.12(b) und (d)).

4.3.4 Bewertung

Bei der Erkennung von rampenförmigen Kanten und schwachen Kontrasten schneidet die Tensorvotierung nicht besser ab als der Canny-Operator oder das Strukturtenor-Verfahren. Im Gegenteil: die Entstehung der Artefakte ist sogar von Nachteil. Sie lässt sich durch die Symmetrie und Regelmäßigkeit der Testfiguren begründen und ist in echten Anwendungen auf Grauwertbildern vermutlich zu vernachlässigen, weil die Artefakte durch die komplexeren, unregelmäßigeren Muster nicht so leicht entstehen können. Bei starkem Rauschen („Shot Noise“ und Gauß'sches Rauschen) liefert die Tensorvotierung bessere Ergebnisse als die anderen beiden Verfahren, was auf die Gruppierungsfähigkeit zurückzuführen ist, die vorherrschende Strukturen verbinden und herausarbeiten kann.

Mögliche Erweiterungen der Tensorvotierung (siehe Kapitel 6), insbesondere die Berücksichtigung von Krümmungsinformationen, können eventuell die Entstehung von Artefakten vermindern und so diesen Nachteil teilweise ausgleichen.

4.4 Zusammenfassung

In diesem Kapitel wurde anhand mehrerer Experimente einzelne Aspekte der Tensorvotierung untersucht. Die Ergebnisse lassen sich folgendermaßen zusammenfassen:

- Die Tensorvotierung ist relativ unempfindlich gegenüber der Größe einer Struktur, kommt gut mit sich überlagernden und nahe beieinander liegenden Strukturen zurecht und arbeitet unabhängig von deren Orientierung. Der Skalenfaktor der Votierfelder spielt dabei eine untergeordnete Rolle.
- Die Tensorvotierung bringt Vorteile bei Bildern mit Rauschen und kann kleine Unebenheiten und Konkavitäten glätten und ausgleichen. Ein relativ großer Skalenfaktor wirkt sich hier positiv auf die Ergebnisse aus.
- Die Tensorvotierung liefert bei der Erkennung von rampenförmigen Kanten und schwachen Kontrasten keine besseren Ergebnisse als der Canny-Operator oder das Strukturtenor-Verfahren.
- Bei den meisten Versuchen entstehen durch die Tensorvotierung unerwünschte Artefakte, die bei größeren Votierfeldern zunehmen.

Konturdetektion bei Herzbildern

Nachdem in den vorangegangenen Kapiteln alle notwendigen Grundlagen für die Extraktion von Kurven aus Grauwertbildern beschrieben worden sind, erläutert dieses Kapitel nun den gesamten Konturfundungsprozess bei den Herzbildern. Im Vordergrund stehen dabei die weiteren Verarbeitungsschritte nach der Extraktion von einzelnen Kontursegmenten. Das Kapitel beginnt mit einer kurzen Übersicht über die Anatomie des Herzens. Es folgt in Abschnitt 5.2 eine Erläuterung der besonderen Problematik und in Abschnitt 5.3 eine Übersicht über das gesamte Verfahren. Anschließend werden die einzelnen Verarbeitungsmethoden ausgeführt.

5.1 Anatomie des Herzens

Thews, Mutschler und Vaupel [TMV99] bieten eine gute Einführung in die Anatomie (Körperbau), Physiologie (Funktionsweise) und Pathophysiologie (krankhafte Vorgänge) des Menschen.

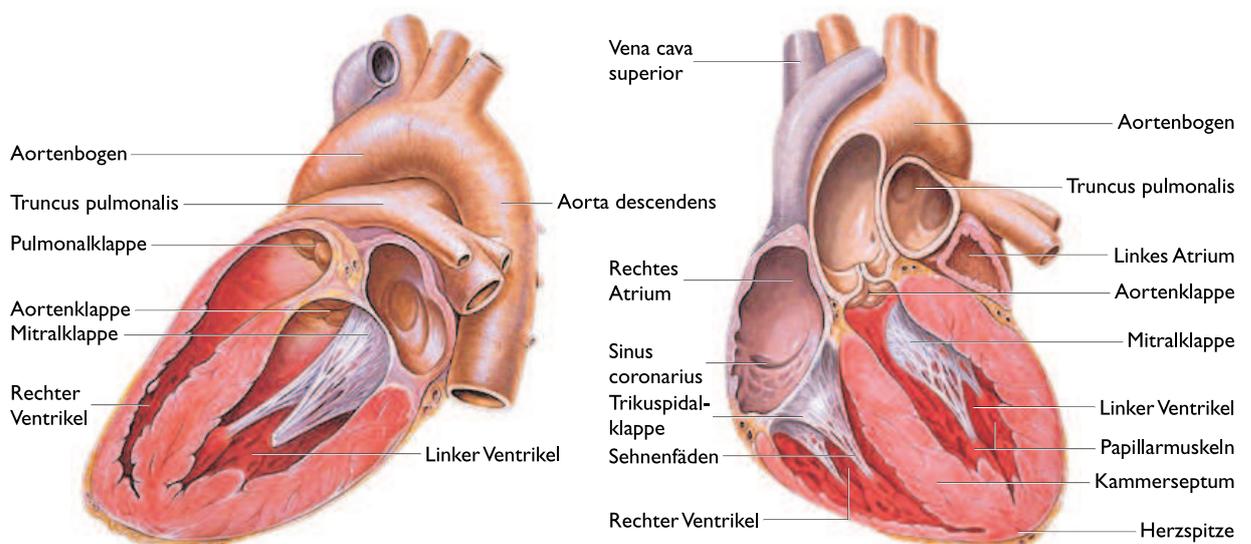


Abbildung 5.1: Die Anatomie des Herzens (Ansicht seitlich und von vorne). (Die Zeichnungen stammen von Patrick J. Lynch [Lyn00] und sind mit Erlaubnis wiedergegeben.)

Hier sei ein kurzer Abriss der Anatomie des Herzens zur Einführung der Terminologie und zum besseren Verständnis wiedergegeben.

Das Herz liegt im vorderen unteren Teil des Mittelfellraums (*Mediastinum*) und wird beiderseits von den äußeren Lungenflügeln begrenzt. Unten stößt es teilweise an das Zwerchfell, vorn an das Brustbein (*Sternum*) und die anschließenden Rippenknorpel und hinten an die Speiseröhre (*Ösophagus*) und andere Organe des Mediastinums. Wegen der engen Beziehungen zum Atmungsapparat ändert das Herz seine Lage mit den Atmungsbewegungen des Zwerchfells, der Rippen und der Lungenflügel.

Man kann das Herz in zwei funktionelle Teile gliedern, das rechte und das linke Herz (siehe Abbildung 5.1 und 5.2). Beide bestehen aus einem kleineren Vorhof (*Atrium*) und einer größeren Kammer (*Ventrikel*), die von unterschiedlich dicken Muskelschichten (*Myokard*) umschlossen sind. Durch rhythmische Kontraktion und Erschlaffung der Herzmuskulatur wird das Blut aus den Hohlvenen (*Vena cava superior* und *inferior*) im rechten Herz bzw. aus den Lungenvenen (*Venae pulmonales*) im linken Herz in die Vorhöfe und von dort in die Kammern gepumpt. Aus den Herzkammern gelangt das Blut in der rechten Herzhälfte in die Lungenschlagader (*Truncus pulmonalis*) und in der linken Herzhälfte in die Hauptschlagader (*Aorta*). Die Herzklappen haben dabei eine Ventilwirkung und verhindern den Rückfluss des Blutes.

5.2 Besondere Problematik

Wie bereits in der Einleitung in Kapitel 1 erwähnt handelt es sich bei den Eingabebildern um MRT-Aufnahmen über den gesamten Herzzyklus hinweg (von der Diastole, in der die Muskeln erschlafft sind, über die Systole, in der die Muskeln sich zusammenziehen, wieder hin zur Diastole). Während der Aufnahmen muss der Patient den Atem anhalten, um Bewegungsartefakte zu vermeiden. Die Aufnahmen werden dann durch das EKG¹ kontrolliert, damit die Bildfolgen jeweils dieselben Phasen des Herzzyklus darstellen [SB01]. Zwei Beispiele für solche Aufnahmen

¹Das Elektrokardiogramm (EKG) ist ein Kurvenbild, das die elektrischen Potentiale bzw. Potentialdifferenzen misst, die bei der Erregungsausbreitung und -rückbildung im Herzen entstehen.

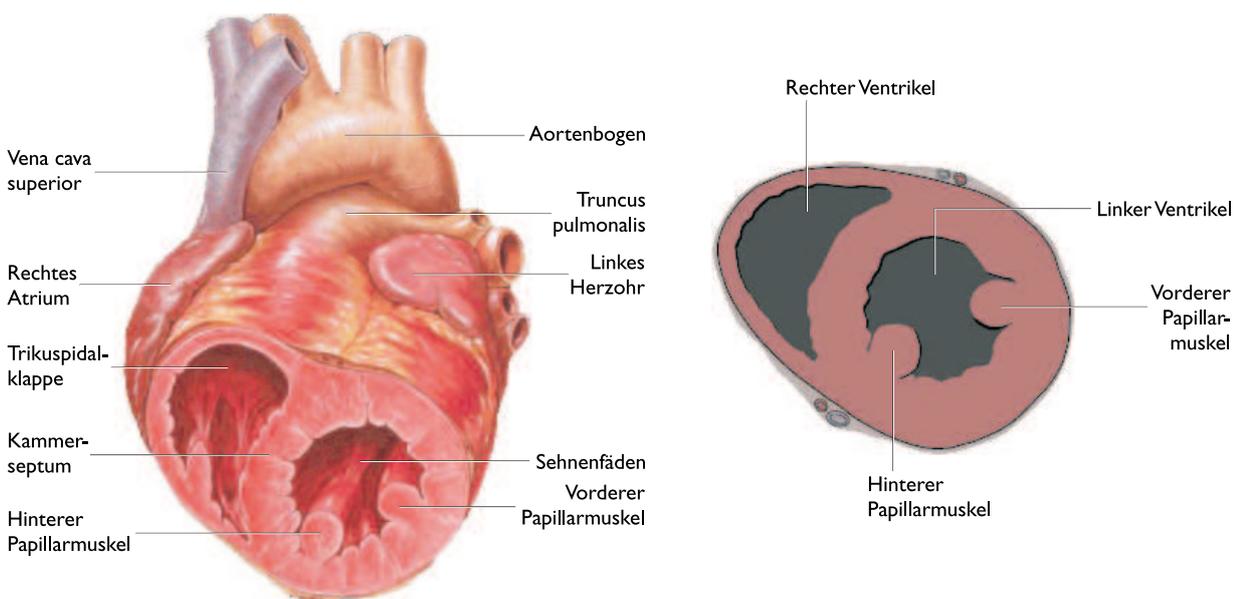


Abbildung 5.2: Die Anatomie des Herzens (Ansicht der Kurzachse und schematisch). (Die Zeichnungen stammen von Patrick J. Lynch [Lyn00] und sind mit Erlaubnis wiedergegeben.)

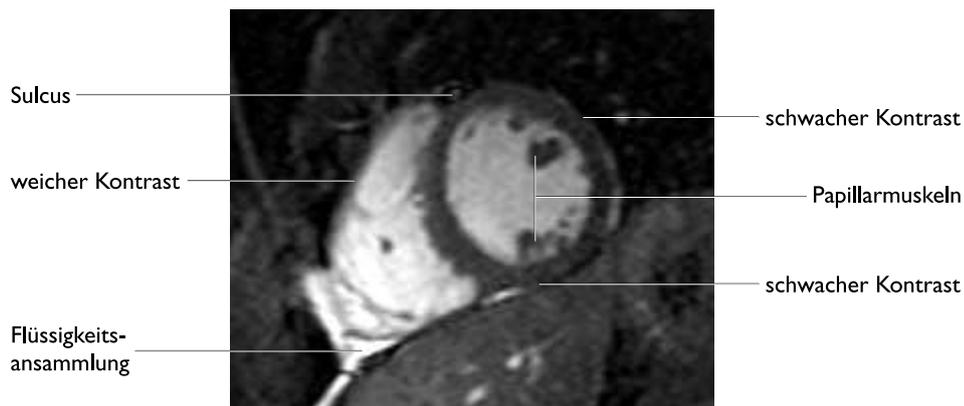


Abbildung 5.3: Problematische Bereiche bei der Konturfindung.

waren bereits in Abbildung 1.1 zu sehen.

Im folgenden ist eine Vielfalt von Aspekten aufgeführt, die die Verarbeitung der Bilder erleichtern oder erschweren und teilweise eine gesonderte Berücksichtigung erfordern. Als vorteilhaft erweisen sich:

- ⊕ Die Bilder zeigen gleiche Ansichten, d. h. gleiche Schnitte durch den betreffenden Brustbereich (hilft Parameter einzustellen und Prozess zu automatisieren).
- ⊕ Die Ventrikel sind im Bild zentriert (hilft bei der Auswahl der Kontursegmente).
- ⊕ Die Ventrikel sind helle Regionen (ermöglicht Verwendung einer Intensitätsmaske).
- ⊕ Die Konturen des linken Ventrikels sind kreisförmig (hilft bei der Konturanpassung und Glättung).
- ⊕ Über mehrere Phasen hinweg bewegt sich fast nur das Herz (ermöglicht Verwendung einer Varianzmaske).
- ⊕ In medizinischen Bildern kommen praktisch keine Ecken und nur wenige Kreuzungspunkte vor (Kreuzungssalienzkarte muss nicht ausgewertet werden).

Nachteilig wirken sich diese Punkte aus:

- ⊖ Die Papillarmuskeln müssen ausgespart werden. Erschwert wird dies dadurch, dass diese in der Schichtaufnahme sowohl als isolierte Inseln als auch als Einstülpungen vom Myokard erscheinen können.
- ⊖ Die Konturen des rechten Ventrikels variieren in der Form. Sie sind in der Regel konvexbogig, haben aber eine größere Varianz als die kreisförmigen Konturen des linken Ventrikels.
- ⊖ Teilweise treten weiche Kontraste auf (rampenförmige Kanten oder „ramp edges“), die eine genaue Lokalisation der Kurve erschweren.
- ⊖ Manche Kontraste sind sehr schwach oder fehlen ganz (oft bei den äußeren Konturen, siehe Abbildung 5.3).
- ⊖ Die inneren und äußeren Konturen des rechten Ventrikels sind wegen der relativ dünnen Muskelschicht fast nicht zu unterscheiden. Anders als beim linken Ventrikel gibt es nur einen dünnen oder gar keinen, abgrenzenden, dunklen Kontrast.

5 Konturdetektion bei Herzbildern

- ⊖ Am tiefsten Punkt in der rechten Herzhälfte kann sich etwas Flüssigkeit ansammeln² (bildet in Abbildung 5.3 ein „Dreieck“), die nur schwer abzugrenzen ist, weil sie genauso hell erscheint wie die rechte Herzkammer und deren Wand sehr dünn ist.
- ⊖ Flüssigkeitsansammlungen und epikardiales Fett in den Sulci zwischen den Herzkammern³ können falsche Hinweise für die äußeren Konturen liefern. Die Konturen verlaufen auf der „Innenseite“ und nicht auf der „Außenseite“ der Sulci (siehe Abbildung 1.1, rechtes Bild).
- ⊖ Die Helligkeit des Myokards schwankt leicht über den Herzzyklus hinweg und wird während der Kontraktion heller (wegen der erhöhten Durchblutung). Dadurch verringern sich die Kontraste zu den Kammern, jedoch sind die Auswirkungen nahezu vernachlässigbar.

5.3 Die Konturextraktion im Überblick

Das Verfahren zur Konturfindung der Herzkammern kann in drei Stufen unterteilt werden (siehe Abbildung 5.4):

1. die Vorverarbeitung der Eingabebilder
2. die Extraktion von Kurvensegmenten mit Hilfe von Strukturtenoren und dem Tensor Voting Verfahren; eine sogenannte Varianzmaske hilft, den Suchraum und damit den Berechnungsaufwand einzuschränken
3. die Auswahl der gesuchten Kurvensegmente und anschließende Vervollständigung zu durchgängigen Kurven und Glättung; für diesen Schritt ist zwangsweise Modellwissen notwendig, d. h. Wissen über Lage, Form und Beschaffenheit bestimmter anatomischer Strukturen

5.4 Vorverarbeitung

Es ist sinnvoll, zu Beginn der Verarbeitung eines Bildes die Qualität allgemein oder in den Regionen, die von Interesse sind, zu verbessern, beispielsweise durch Anpassung von Helligkeit und Kontrast oder Beseitigung von Rauschen durch Anwendung eines Tiefpassfilters. Dadurch können die Ergebnisse der nachfolgenden Verarbeitungsstufen verbessert werden.

Tatsächlich wird der Kontrast der Eingabebilder automatisch durch eine Grauwert-Transformation verbessert. Dies geschieht, indem man zuerst ein Histogramm berechnet, dann ein Teil der dunkelsten Werte (z.B. 1%) auf Schwarz und ein Teil der hellsten Werte (ebenfalls 1%) auf Weiß setzt und schließlich die übrigen Grauwerte auf den gesamten, möglichen Wertebereich streckt. Durch diese Methode werden die möglichen Intensitätswerte besser ausgenutzt und bei Eingabebildern mit schlechtem Kontrast erzielt man so bessere Ergebnisse.

Auf die Anwendung eines Tiefpassfilters kann in der Vorverarbeitung verzichtet werden, da zur Berechnung der Strukturtenoren eine Glättung stattfindet und die anschließende Tensorvoting ebenfalls sehr gut mit Rauschen umgehen kann (siehe Kapitel 4).

Da nur die Konturen der Herzkammern gesucht sind, werden die Bilder für die weitere Verarbeitung beschnitten. Auf diese Weise verringert sich der Berechnungsaufwand, denn die außen liegenden Bereiche, deren Ergebnisse sowieso nicht relevant sind, müssen nicht verarbeitet werden. Die extrahierten Konturen können am Ende des Verfahrens wieder in die originalen Bilder übertragen werden.

²Dies muss nicht krankhaft sein und kommt auch bei gesunden Organen vor.

³Hier sind die Furchen gemeint, in denen die Herzkranzgefäße verlaufen.

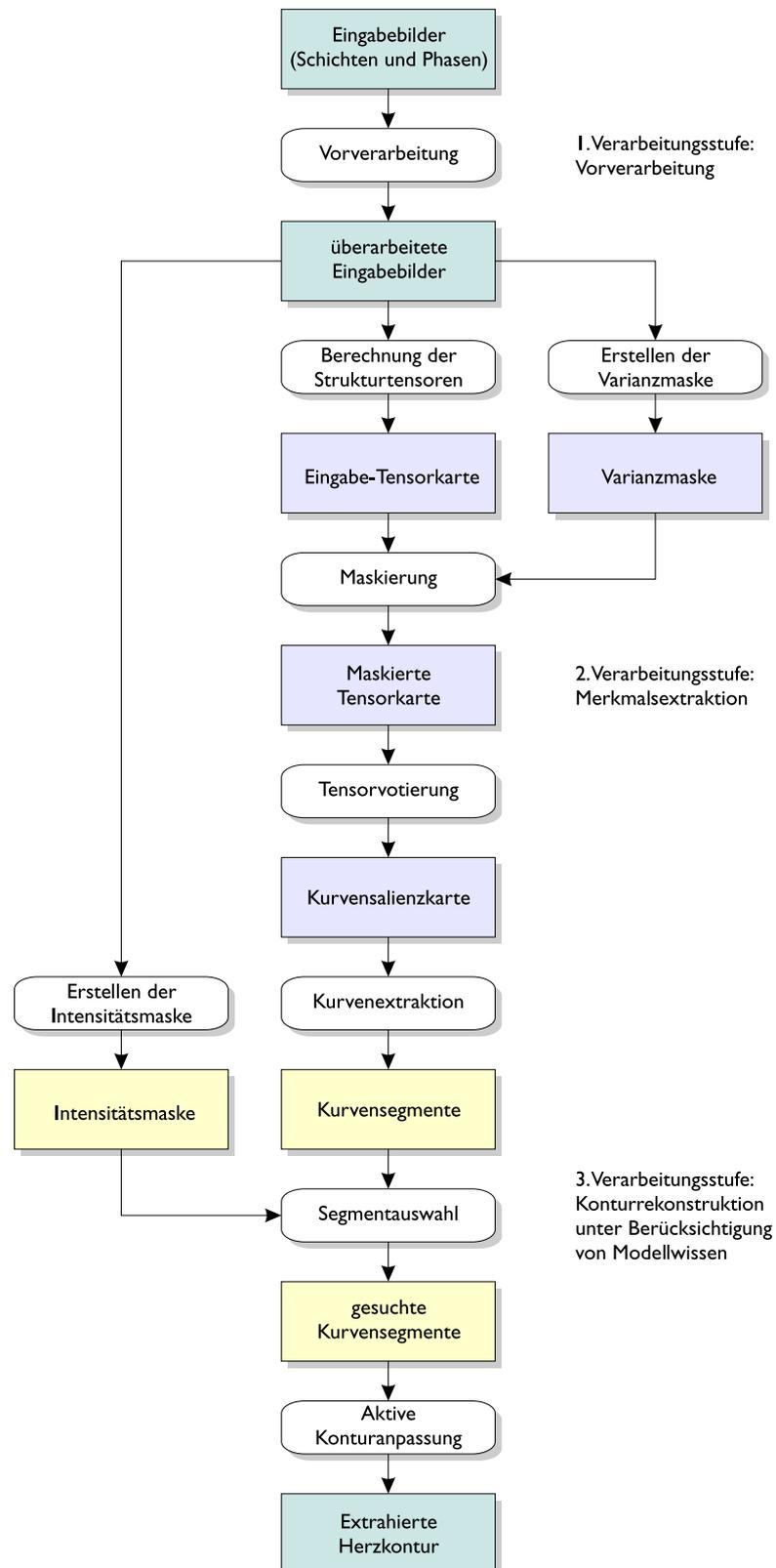


Abbildung 5.4: Übersicht über das Verfahren zur Extraktion der Herzkonturen. Die Verarbeitungsstufen sind farblich unterschiedlich gekennzeichnet.

5.5 Extraktion von Kurvensegmenten

Das im vorigen Schritt aufbereitete Eingabebild muss nun mit Hilfe des Tensor Voting Verfahrens weiterverarbeitet werden, um Kurvensegmente für die nachfolgende Auswahl und Konturanpassung zu erhalten. Wie in Kapitel 3 beschrieben werden zuerst die Strukturensoren berechnet und dann – als Optimierung – eine Varianzmaske ermittelt und darauf angewendet. Schließlich führt man eine Tensorvotierung durch und extrahiert die Kurvensegmente aus der Kurvensalienzkarte. Die Kreuzungssalienzkarte muss nicht ausgewertet werden, weil bei dieser Anwendung keine relevanten Kreuzungspunkte vorkommen (siehe Abschnitt 5.2).

5.5.1 Die Varianzmaske

Da sich über die Phasen eines Herzzyklus hinweg hauptsächlich das Herz bewegt (oder genauer: die Herzmuskulatur und das durchströmende Blut), während die umliegenden Organe größtenteils unverändert bleiben, ist es sinnvoll, eine Maske aufgrund der Veränderungen der Bildwerte zu erstellen, eine sogenannte *Varianzmaske*. Die Varianzmaske hilft dabei, den im Bild zu untersuchenden Bereich einzugrenzen, wodurch sich zum einen der Berechnungsaufwand deutlich senkt und zum anderen die Auswahl der gesuchten Kurvensegmente vereinfacht. Diese Idee wurde von Sörgel [Sör96] übernommen, wo sie zur Detektion der Umrisslinie des Herzens eingesetzt wird. Abbildung 5.5 zeigt ein Varianzbild, das aus einer zeitlichen Folge von Herzbildern berechnet wurde, und die daraus erstellte Varianzmaske.

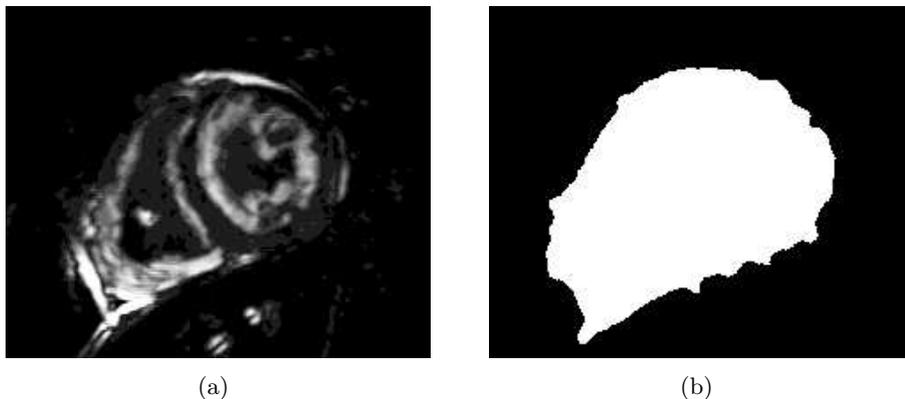


Abbildung 5.5: Varianzbild und Varianzmaske. Je heller ein Bereich des Varianzbildes ist, umso mehr verändern sich die Grauwerte über die Zeit hinweg. Das Varianzbild wurde stark aufgehellt,⁵ um die Bereiche mit höherer Varianz, also die „bewegten“ Bereiche, sichtbar zu machen.

Nachfolgend wird schrittweise die Erzeugung der Varianzmaske erläutert:

1. Berechnung des Mittelwertes aus allen Bildern $g_t(x, y)$ eines Zyklus ($t = 1, 2, \dots, T$),

$$\bar{g}(x, y) = \frac{1}{T} \sum_{t=1}^T g_t(x, y)$$
2. Berechnung der Varianz $V_g(x, y) = \frac{1}{T-1} \sum_{t=1}^T (g_t(x, y) - \bar{g}(x, y))^2$
3. Einteilung der Werte der Varianz $V_g(x, y)$ in k Klassen und Berechnung des zugehörigen Histogramms $H(v)$, wobei v die Obergrenze einer Histogrammklasse angibt.

⁵Durch eine Histogrammlinierisierung und anschließende Gammawert-Korrektur.

4. Schwellwert T_v ermitteln, indem für T_v der größte Wert gesucht wird, für den die folgende Ungleichung gilt (T_v kann dabei dieselben Werte annehmen wie v):

$$\int_{T_v}^1 H(v) \, dv \leq T_H \quad (5.1)$$

wobei T_H ein empirisch (z. B. von einem Arzt) festgelegter Schwellwert ist, der in etwa die Größe der Herzregion in Pixeln angibt. Der Schwellwert hängt demnach stark von den Aufnahmeparametern sowie der Bildgröße und -auflösung ab. Für die Maske in Abbildung 5.5 wurde ein Wert von $T_H = 30000$ gewählt.

5. Berechnung der Maske V_m durch Anwendung des Schwellwertes T_v :

$$V_m(x, y) = \begin{cases} 1, & V_g(x, y) > T_v \\ 0, & V_g(x, y) \leq T_v \end{cases} \quad (5.2)$$

6. Anwendung eines morphologischen Operators [GW01] zur Entfernung kleiner Störungen, wie z. B. „Shot Noise“ (siehe Abschnitt 4.2.2), und Glättung der Ränder der Regionen:

$$V_m(x, y) = \begin{cases} 1, & \sum_{(u,v) \in N(x,y)} V_m(u, v) > N_{\max} \\ 0, & \sum_{(u,v) \in N(x,y)} V_m(u, v) < N_{\min} \\ V_m(x, y), & \text{sonst} \end{cases} \quad (5.3)$$

wobei $N(x, y)$ eine Nachbarschaft um einen Punkt (x, y) angibt, z. B. eine $N \times N$ Maske, und N_{\max} bzw. N_{\min} zwei Schwellwerte sind. Mit $N = 7$, $N_{\max} = 24$ und $N_{\min} = 20$ schließt der Operator Löcher mit bis zu 24 Pixeln und entfernt kleine Objekte mit bis zu 19 Pixeln.

7. Nur die größte zusammenhängende Region erhalten (alle anderen unterdrücken)
8. Löcher innerhalb der Region ausfüllen (alle Bildpunkte, die nicht vom Hintergrund aus über eine Vierer- oder Achternachbarschaft erreichbar sind, werden ausgefüllt)

Die Varianzmaske muss *nach* der Berechnung der Strukturtenoren angewendet werden, weil sonst harte Kontraste an den Rändern des maskierten Bereichs entstehen können, die als unerwünschte Konturen erkannt werden würden.

5.6 Auswahl der Kurvensegmente

Aus der Menge der zahlreichen Kurvensegmente, die in der zweiten Verarbeitungsstufe gefunden wurden, müssen nun die gesuchten Segmente ausgewählt werden. Außer den gewünschten Segmenten befinden sich nämlich in der zu untersuchenden Menge sowohl Kurvensegmente anderer Strukturen als auch Artefakte. Zur Auswahl bieten sich unterschiedliche Strategien an, jedoch erwies sich die Verwendung einer Intensitätsmaske als sehr vielversprechend. Andere Möglichkeiten sind:

- *Verwerfen kurzer Linienstücke.* Dieser Ansatz bietet sich nur zur Verringerung der Anzahl der zu untersuchenden Kontursegmente an, jedoch ist die Länge kein Kriterium, gewünschte von unerwünschten Segmenten zu unterscheiden, denn eine Kontur kann mehrfach unterteilt sein und kurze Segmente enthalten.

- *Verwerfen von Linien mit geringer Krümmung.* Zumindest die beiden Konturen des linken Ventrikels weisen eine größere Krümmung auf, so dass man theoretisch alle Segmente mit einer geringen Krümmung verwerfen könnte. Allerdings bringt auch dieser Ansatz keine Vorteile, denn kurze Segmente der gesuchten Konturen weisen oft eine kleine Krümmung auf, außerdem kann die Kontur des rechten Ventrikels gerade Stücke enthalten. So würde man diese Segmente fälschlicherweise verwerfen.

Letztlich wird wegen der allgemeinen Problematik für jede gesuchte Kontur eine spezielle Methode zur Auswahl der entsprechenden Segmente angewendet. Diese Methoden werden in den folgenden Abschnitten genauer erläutert.

5.6.1 Intensitätsmaske

Sehr hilfreich bei der Lokalisation der Herzkammern ist die Tatsache, dass sie als sehr helle und große Regionen im Bild dargestellt sind. Diesen Umstand nutzt man, um eine sogenannte *Intensitätsmaske* zu erstellen, eine binäre Maske beruhend auf den Intensitätswerten des Bildes. Mit ihrer Hilfe kann man die inneren Konturen und (Masse)schwerpunkte der beiden Herzkammern schätzen.

Die Intensitätsmaske wird ähnlich der Varianzmaske (s. o.) konstruiert:

1. Normalisierung der Grauwerte des Bildes: $g(x, y) = \frac{g(x,y)}{\max_{x,y} g(x,y)}$.
2. Anwendung eines Schwellwertes T_I , um dunklere Bereiche zu unterdrücken:

$$b(x, y) = \begin{cases} 1, & g(x, y) > T_I \\ 0, & g(x, y) \leq T_I \end{cases} \quad (5.4)$$

3. Anwendung des morphologischen Operators aus Gleichung 5.3 zur Bereinigung der Maske
4. Alle Regionen außer den beiden größten verwerfen
5. Löcher innerhalb der beiden Regionen ausfüllen

Die Intensitätsmaske ist eine gute Heuristik, um erwünschte von unerwünschten Kurvensegmenten zu trennen. Die größten Probleme sind dabei die Aussparung der Papillarmuskeln und die

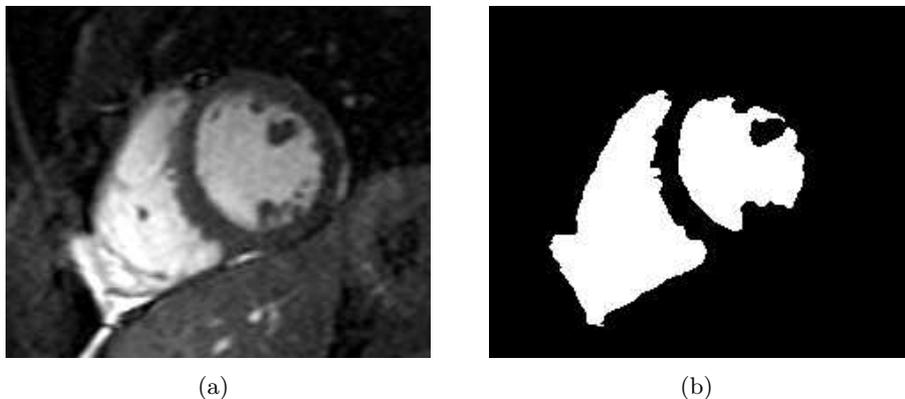


Abbildung 5.6: Originalbild und Intensitätsmaske ($T_I = 0.4$). Bei der linken Herzkammer bleiben die Papillarmuskeln ausgespart, bei der rechten wurde die Flüssigkeitsansammlung (links unten) fälschlicherweise hinzugenommen.

Hinzunahme von Flüssigkeitsansammlungen (siehe Abbildung 5.6), beides ungewollte Eigenschaften, die dazu führen, dass in diesen Bereichen in den nachfolgenden Schritten falsche Segmente ausgewählt werden. Für die meisten Segmente jedoch funktioniert die Auswahl sehr gut.

5.6.2 Endokontur des linken Ventrikels

Die Kurvensegmente der inneren Kontur der linken Herzkammer können folgendermaßen bestimmt werden (siehe auch Abbildung 5.7):

1. rechte Region in der Intensitätsmaske auswählen (z. B. durch Vergleich der Schwerpunkte der beiden Regionen in der Intensitätsmaske)
2. Umrisslinie der ausgewählten Region ermitteln (z. B. durch einen binären morphologischen Operator)
3. alle Kurvensegmente auswählen, die innerhalb einer Nachbarschaft der Umrisslinie liegen (innerhalb einer Entfernung von N Pixeln). Dabei können Segmente weiter unterteilt werden, wenn sie nur teilweise innerhalb der Nachbarschaft liegen.

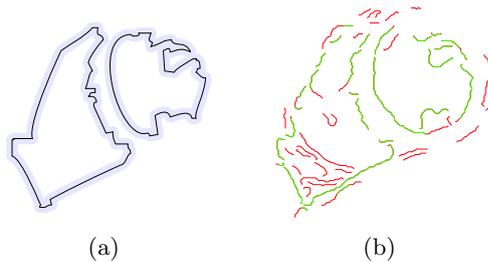


Abbildung 5.7: Auswahl der inneren Konturen mit Hilfe der Intensitätsmaske. (a) Umrisslinien der Intensitätsmaske aus Abbildung 5.6 mit eingezeichneter Nachbarschaft. (b) Auswahl der Kurvensegmente innerhalb der Nachbarschaft (grün).

5.6.3 Epikontur des linken Ventrikels

Die Auswahl der Kurvensegmente für die äußere Kontur der linken Herzkammer erweist sich als etwas schwieriger als diejenige für die innere Kontur. Folgende Schritte sind notwendig (siehe Abbildung 5.8):

1. Schwerpunkte der beiden Regionen in der Intensitätsmaske berechnen (Abb. 5.8(a))
2. Profil der Intensitätswerte auf der Verbindungslinie der beiden Schwerpunkte ermitteln (Abb. 5.8(b))
3. Die Positionen des starken Intensitätsabfalls und -anstiegs ermitteln. Sie entsprechen dem Anfang und Ende des Kammerseptums und werden bestimmt, indem man entlang des ausgelesenen Profils vom Schwerpunkt des rechten Ventrikels zum Schwerpunkt des linken Ventrikels wandert und nach einem globalen Minimum sucht. Anfang und Ende des Kammerseptums werden durch einen Schwellwert festgelegt, d. h. der Anfang entspricht der Position, an der der Intensitätswert unter den Schwellwert sinkt, das Ende der Position, an der der Intensitätswert über den Schwellwert ansteigt. Die Endposition des Kammerseptums wird dann mit der Position der äußeren Kontur des linken Ventrikels gleichgesetzt.

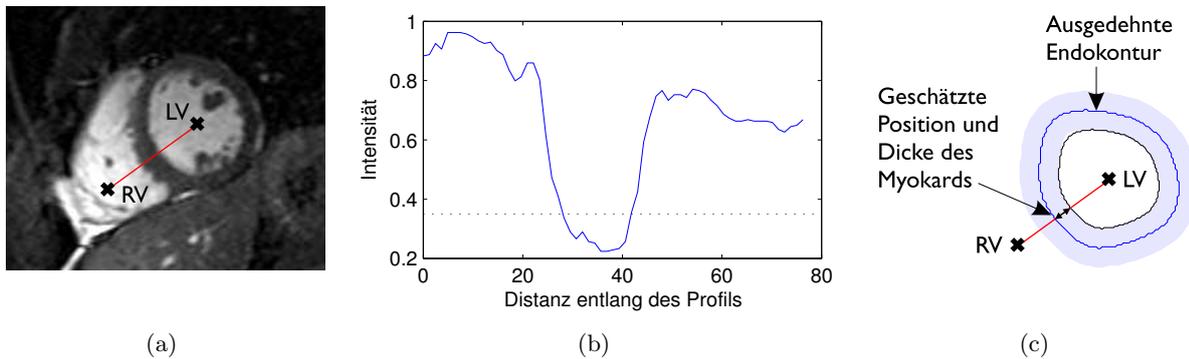


Abbildung 5.8: Auswahl der äußeren Kontur des linken Ventrikels. (a) zeigt die aus der Intensitätsmaske ermittelten Schwerpunkte der beiden Ventrikel und deren Verbindungslinie, aus der die Intensitätswerte ausgelesen werden. In dem daraus entstandenen Profil (b) ist deutlich Anfang und Ende des Kammerseptums zu sehen (starker Abfall bzw. Anstieg). Daraus wird die Position der äußeren Kontur geschätzt und eine Nachbarschaft bestimmt (c), aus der die Kurvenssegmente für die Konturfindung ausgewählt werden.

4. Endokontur auf Endposition des Kammerseptums ausdehnen und alle Kurvenssegmente innerhalb einer Nachbarschaft auswählen. Als Ausdehnung der Nachbarschaft wird die im vorigen Punkt geschätzte Dicke des Kammerseptums verwendet (Abb. 5.8(c)).

5.6.4 Konturen des rechten Ventrikels

Die Konturen der rechten Herzkammer sind in vielen Phasen des Herzzyklus nur schwer oder gar nicht zu unterscheiden, weil die im Vergleich zur linken Herzkammer deutlich dünnere Muskelschicht kaum zu erkennen ist (siehe Abschnitt 5.2). Wegen dieser erschwerten Bedingungen wurde die Konturfindung auf den inneren Umriss beschränkt; der äußere Umriss wird in dem vorgestellten Verfahren nicht extrahiert.

Die Kurvenssegmente der inneren Kontur werden entsprechend dem linken Ventrikel ausgewählt, d. h. zuerst wird die Umrisslinie der Intensitätsmaske gebildet und dann alle Segmente innerhalb einer Nachbarschaft ausgewählt (siehe Abschnitt 5.6.2).

5.7 Gruppierung der Kurvenssegmente

Im letzten Schritt der Konturfindung müssen die einzelnen, ausgewählten Kurvenssegmente zusammengefasst und verbunden werden, so dass kontinuierliche Kurven entstehen. Die Kurven müssen dann eventuell geglättet werden, falls sie noch zu stark schwanken. Es gibt unterschiedliche Ansätze für diese Gruppierung der Kurvenssegmente, beispielsweise eine Spline-Interpolation [FDFH95], „Active Contour Models“ und das darauf basierende „Gradient Vector Flow“ Verfahren [XP98] oder eine Gruppierung mittels Tangenten und einer Graphendarstellung nach Elder und Zucker [EZ96]. Hier wurde der „Gradient Vector Flow“ Ansatz gewählt, weil er direkt mit zusammenhängenden und geschlossenen Konturen arbeitet und relativ gut mit den Einbuchtungen der Papillarmuskeln umgehen kann.

5.7.1 Gradient Vector Flow

An dieser Stelle sei nur kurz das Prinzip des „Gradient Vector Flow“ Verfahrens beschrieben; weitere Details sind beispielsweise in [XP98] zu finden. „Active Contour Models“ oder *Snakes* kann man sich als Kurven vorstellen, die einem elastischen Band ähneln. Sie bestehen aus einer Menge von Punkten, die durch Linien zu einer geschlossenen Kreisform verbunden sind. Diese Form gehorcht sogenannten *inneren* und *äußeren Kräften* und passt sich ihnen Schritt für Schritt an, solange bis sie sich ausgleichen. Mathematisch wird dies durch die Minimierung eines Funktionals ausgedrückt (siehe Abschnitt 1.2.2). Die innere Kraft wird hierbei durch *Elastizität*, *Rigidität* und *Viskosität* beschrieben, während die äußere Kraft dem Vektorfeld eines Bildgradienten entspricht. Das „Gradient Vector Flow“ Verfahren unterstützt als Erweiterung außerdem noch eine dritte Kraft, den nach außen gerichteten *Druck* („pressure“). Das Verfahren wird mit einer anfänglich vorgegebenen Kontur gestartet, lässt dann die Kräfte auf die Kontur einwirken und aktualisiert so die Positionen aller Punkte. Dies wiederholt sich mit entsprechend anderen Kräften an den neuen Positionen, bis sich die Punkte praktisch nicht mehr ändern.

Das Vektorfeld für die äußere Kraft wird folgendermaßen erzeugt: zuerst wird ein binäres Bild erstellt, in dem nur die ausgewählten Kurvensegmente dargestellt sind. Das Bild wird dann mit einem Gauß-Filter stark geglättet und danach werden die Grauwertgradienten des geglätteten Bildes und damit die äußere Kraft berechnet. Die Glättung dient dazu, den Einflussbereich der äußeren Kraft zu vergrößern. Die Werte für alle anderen Kräfte – Elastizität, Rigidität, Viskosität und Druck – werden empirisch ermittelt.

Als Vorlage für die innere Kontur dient ein Kreis, dessen Mittelpunkt und Durchmesser aus der Fläche des linken Ventrikels in der Intensitätsmaske bestimmt wird: der Mittelpunkt entspricht dem Schwerpunkt dieser Fläche und der Durchmesser der maximalen Ausdehnung der Fläche zuzüglich einer kleinen Toleranz. Die endgültige Form entsteht dann durch eine schrittweise Schrumpfung und Anpassung der Vorlage an das Vektorfeld des Grauwertgradienten.

Als Vorlage für die äußere Kontur dient die innere, die auf die Endposition des Kammerseptums aufgedehnt und dann an die Kurvensegmente angepasst wird (siehe auch Abbildung 5.8(c)).

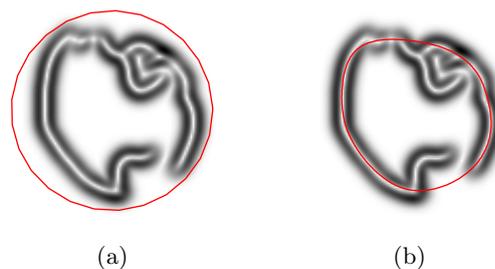


Abbildung 5.9: Gradient Vector Flow für die Endokontur des linken Ventrikels. Dargestellt ist die Stärke des Gradienten, d. h. die Stärke der äußeren Kraft, sowie die Form zu Beginn (a) und am Ende (b) des Verfahrens.

5.8 Ergebnisse

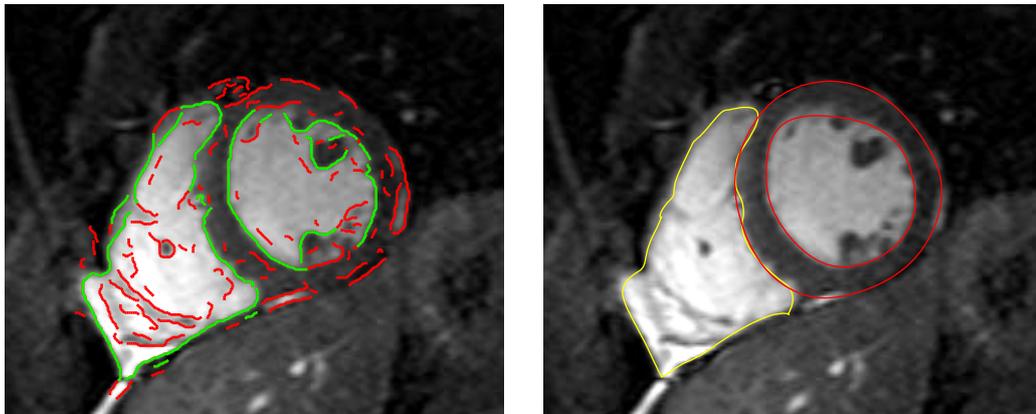
Bei der Auswertung müssen die beiden Verarbeitungsstufen „Tensorvotierung“ und „Auswahl und Gruppierung der Kurvensegmente“ gesondert betrachtet werden, da sich die Verfahren grundsätzlich unterscheiden. Deshalb bezieht sich dieser Abschnitt nur auf die letzte Verarbeitungsstufe, wie sie in diesem Kapitel vorgestellt wurde (die Tensorvotierung wurde bereits ausgiebig in Kapitel 4 betrachtet).

Die Abbildungen 5.10 und 5.11 zeigen einige Ergebnisse der zweiten und dritten Verarbeitungsstufe. Betrachtet man die Konturen der linken Herzkammer, so zeigt sich, dass diese gut erkannt werden. Es gibt jedoch auch einige, inakzeptable Ausnahmen (zu sehen in Abb. 5.10(b)), die z. B. aufgrund schlecht eingestellter Parameter bei der aktiven Konturanpassung entstehen können. Problematisch sind auch die mittleren Phasen des Herzzyklus, da während der Kontraktion des Herzmuskels die Papillarmuskeln im Verhältnis mehr Platz einnehmen und so den Ausgleich und die Glättung der Kontur erschweren.

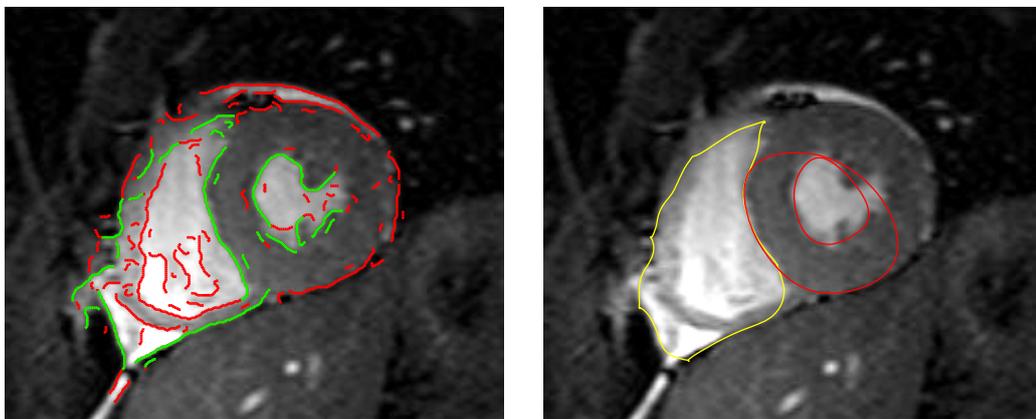
Die Extraktion der Konturen des rechten Ventrikels versagt in den meisten Fällen, was in erster Linie an der ungeeigneten Auswahl der Kurvensegmente durch die Intensitätsmaske liegt. Dazu ist zu sagen, dass bei der Entwicklung der Konturfindungsverfahren der Schwerpunkt auf der Extraktion der Konturen der linken Herzkammer lag und die Konturen der rechten Herzkammer nachrangig und nur unzureichend behandelt wurden.

Insgesamt weist das Verfahren folgende Stärken und Schwächen auf:

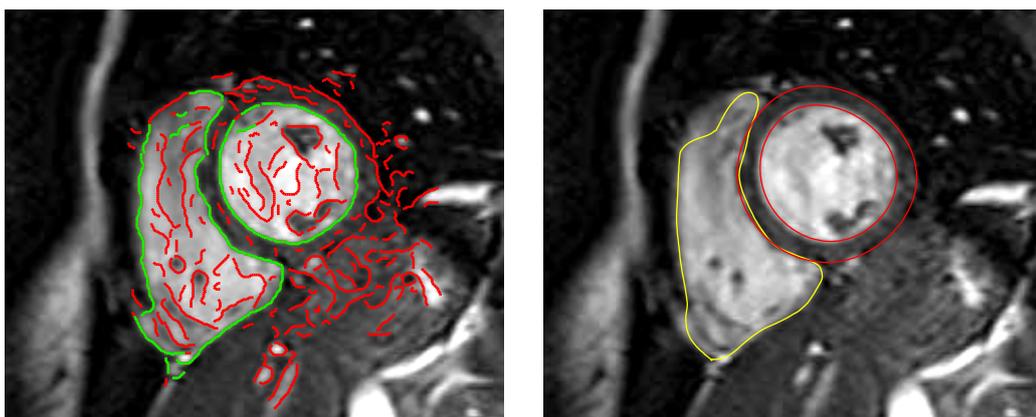
- ⊕ Die aktive Konturanpassung kann sehr einfach einzelne Kurvensegmente verbinden und in der Regel die Einbuchtungen der Papillarmuskeln sehr gut ausgleichen.
- ⊕ Die Konturen werden gut lokalisiert (sofern im ersten Schritt die „richtigen“ Kurvensegmente ausgewählt werden).
- ⊖ Die Intensitätsmaske spart die Papillarmuskeln und andere dunkle Bereiche aus, wodurch unerwünschte Kontursegmente ausgewählt werden und verzerrte Endokonturen entstehen können (vgl. Oberkante der Endokontur in Abbildung 5.10(a)).
- ⊖ Die Einstellung der Parameter des „Gradient Vector Flow“ Verfahrens ist sehr schwierig und wirkt sich empfindlich auf die Ergebnisse aus.
- ⊖ Die Region des rechten Ventrikels in der Intensitätsmaske ist nicht für die Segmentauswahl geeignet. Da die Intensitätswerte innerhalb der rechten Herzkammer stärker schwanken als innerhalb der linken, weicht die maskierte Form von der tatsächlichen teilweise erheblich ab. Außerdem bereiten Flüssigkeitsansammlungen und epikardiales Fett Schwierigkeiten (siehe Abschnitt 5.2).



(a) Bildsequenz m0030, Diastole

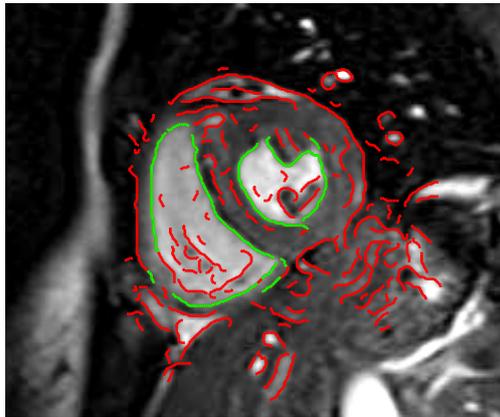


(b) Bildsequenz m0030, Systole

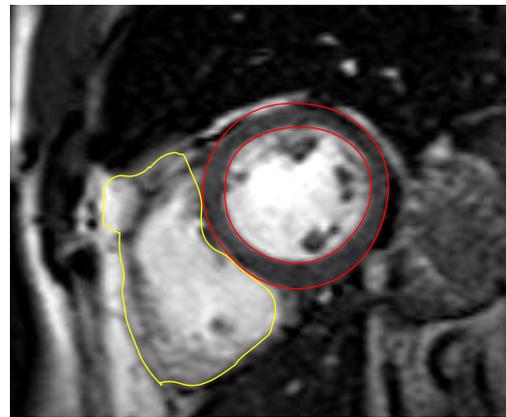
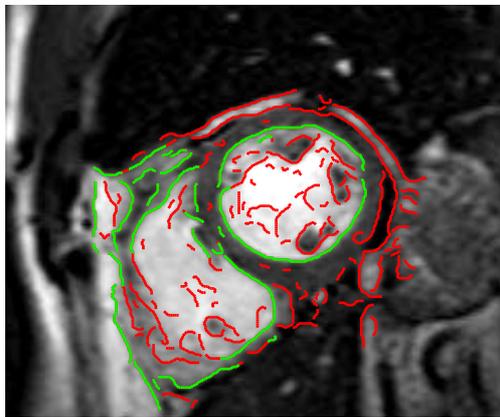


(c) Bildsequenz m0053, Diastole

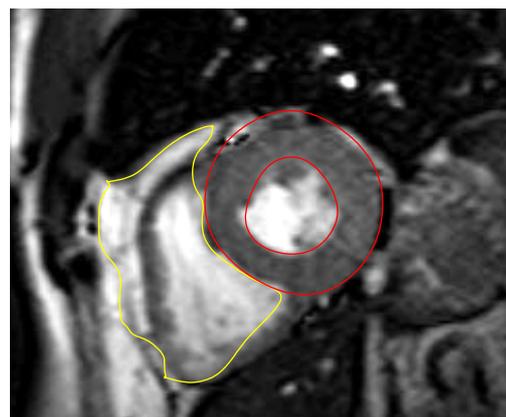
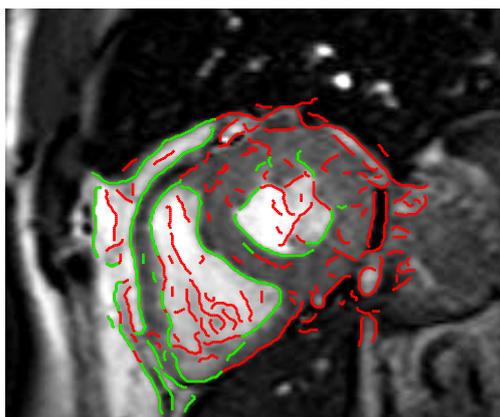
Abbildung 5.10: Beispiele für die Konturextraktion. Aus drei Bildsequenzen wurde jeweils ein Bild während der Entspannung (Diastole) und Anspannung (Systole) des Herzmuskels ausgewertet. Links sind die extrahierten Kurvensegmente dargestellt. Die für die inneren Konturen ausgewählten Segmente sind grün markiert. Rechts sind die gefundenen Konturen der linken (rot) und rechten (gelb) Herzkammer dargestellt. Letztere sind als experimentell und daher nicht als aussagekräftig anzusehen.



(a) Bildsequenz m0053, Systole



(b) Bildsequenz m0233, Diastole



(c) Bildsequenz m0233, Systole

Abbildung 5.11: Beispiele für die Konturextraktion (Fortsetzung). Aus drei Bildsequenzen wurde jeweils ein Bild während der Entspannung (Diastole) und Anspannung (Systole) des Herzmuskels ausgewertet. Links sind die extrahierten Kurvensegmente dargestellt. Die für die inneren Konturen ausgewählten Segmente sind grün markiert. Rechts sind die gefundenen Konturen der linken (rot) und rechten (gelb) Herzkammer dargestellt. Letztere sind als experimentell und daher nicht als aussagekräftig anzusehen.

Dieses Kapitel stellt Erweiterungen für das Tensor Voting Verfahren aus Kapitel 2 vor, die die Qualität der Ergebnisse verbessern oder zusätzliche Informationen liefern sollen. Dafür gibt es unterschiedliche Ansatzpunkte, jeweils mit unterschiedlichem Ziel. Vier Möglichkeiten zur Erweiterung des Verfahrens werden im folgenden kurz vorgestellt. Der erste Abschnitt zeigt, wie man zeitlich benachbarte Bilder einer Bildfolge in der Votierung mit berücksichtigen kann, der zweite, wie man die lokale Krümmung von Eingabemerkmale ermitteln und im Votierdurchgang nutzen kann, der dritte, wie man die Polarität bestimmen und damit Regionen durch eine Umrisslinie begrenzen kann, und der letzte Abschnitt schließlich, wie man das Grundscheema durch mehrere Iterationen abwandeln kann. Die ersten beiden Methoden wurde versuchsweise umgesetzt und ausgetestet, jedoch nicht ausführlich ausgewertet, da dies über den Rahmen der Arbeit hinausgegangen wäre. Die anderen beiden Methoden sind nur der Vollständigkeit halber erwähnt und wurden nicht weiter untersucht.

6.1 Votierung über Phasen des Herzzyklus hinweg

Anstelle der gewöhnlichen Tensorvotierung (aus Kapitel 2), die nur ein einzelnes Bild betrachtet, kann für die gegebene Folge von Herzbildern eine erweiterte Version eingesetzt werden, die mehrere Bilder für die Votierung heranzieht. Ihr Zweck besteht darin, Störungen wie Rauschen und Bewegungsartefakte auszugleichen und die Konturen über die Phasen des Herzzyklus hinweg anzugleichen, damit keine allzu großen Schwankungen entstehen, die für eine spätere Auswertung störend wären.

Der hier vorgestellte Ansatz unterscheidet sich von der Methode von Nicolescu und Medioni [NM02], die zwei aufeinander folgende Bilder betrachtet. Die zwei räumlichen Dimensionen werden dabei um die Zeit erweitert und durch eine vierdimensionale Tensorvotierung die Bewegungen bzw. Geschwindigkeiten in x - und y -Richtung ermittelt. Geschwindigkeiten sind für die Problematik in dieser Arbeit jedoch nicht erforderlich. Deshalb werden in dem vorliegenden Ansatz einfach weiterhin Stimmen im zweidimensionalen Raum gesammelt, allerdings empfängt jedes Bild zusätzlich Stimmen von seinen zeitlich benachbarten Bildern (siehe Abbildung 6.1). Die gesammelten Stimmen werden dabei zeitlich gewichtet mit einer Abschwächungsfunktion

$$TF(t) = e^{-\frac{\tau^2}{w^2}} \quad \text{mit } \tau \in \{-p, \dots, p\} \text{ und } p \in \mathbb{N} \quad (6.1)$$

wobei w einen konstanten Faktor, τ den zeitlichen Abstand und p die Anzahl der zu berücksich-

tigenden, benachbarten Bilder angibt. Der Gewichtungsfaktor w entspricht dem Skalenfaktor σ_v (siehe Abschnitt 2.5.4), bezieht sich jedoch nicht auf die räumliche Entfernung, sondern auf den zeitlichen Abstand. Der zeitliche Abstand τ ist eine ganze Zahl und kann als relativer Index auf die Bilder der Bildfolge gesehen werden. Es wird angenommen, dass eine Bildfolge zyklisch ist, d. h. dass der Vorgänger des ersten Bildes das letzte der Folge und der Nachfolger des letzten Bildes das erste der Folge ist. Enthält die Bildfolge T Bilder, so wird der absolute Index b auf ein benachbartes Bild bei gegebenem Bild $t \in \{1, 2, \dots, T\}$ und zeitlichem Abstand τ folgendermaßen beschrieben:

$$b - 1 \equiv t + \tau - 1 \pmod{T} \quad (6.2)$$

Aus Gleichung 2.7 und 6.2 kann die Gesamtformel für die Votierung über die Phasen des Herzzyklus hinweg hergeleitet werden. Sie ergibt sich für einen *einzelnen* Ort zu einem Zeitpunkt t wie folgt:

$$\mathbf{T}_t^{\text{neu}} = \mathbf{T}_t^{\text{alt}} + \sum_{\tau=-p}^p \left(e^{-\frac{\tau^2}{w^2}} \sum_{k \in N(t)} v(k, t - \tau) \right) \quad (6.3)$$

mit der abgegebenen Stimme:

$$v(k, t) = (\lambda_{1,k}^t - \lambda_{2,k}^t) V_k^t$$

wobei $N(t)$ die Indizes der Tensoren in der Nachbarschaft des zum Zeitpunkt t betrachteten Tensors zurückgibt. V_k^t ist die Stimme des bereits verschobenen und rotierten Stab-Votierfeldes für den empfangenden Tensor. $\lambda_{1,k}^t$ und $\lambda_{2,k}^t$ sind die Eigenwerte des abstimmenden, k -ten Tensors und ihre Differenz entspricht der Stabsalienz. Der Index t bestimmt, von welchem (benachbarten) Bild eine Stimme stammt. Der Index wird dabei entsprechend Gleichung 6.2 angepasst.

Die obige Berechnung muss für jeden Tensor in einem Bild durchgeführt werden. Danach wird er wie gewohnt in sein Eigensystem zerlegt und anschließend ausgewertet (siehe Abschnitt 2.4.3 und 2.4.4).

Durch die Kombination mehrerer zeitlich benachbarter Bilder werden vorherrschende Strukturen verstärkt, während schwächere und zufällig auftretende Strukturen gehemmt werden. Die Berücksichtigung der benachbarten Phasen erhöht allerdings den Berechnungsaufwand beträchtlich: bei Berücksichtigung von p vorausgehenden und p nachfolgenden Bildern müssen $(2p+1)$ -mal

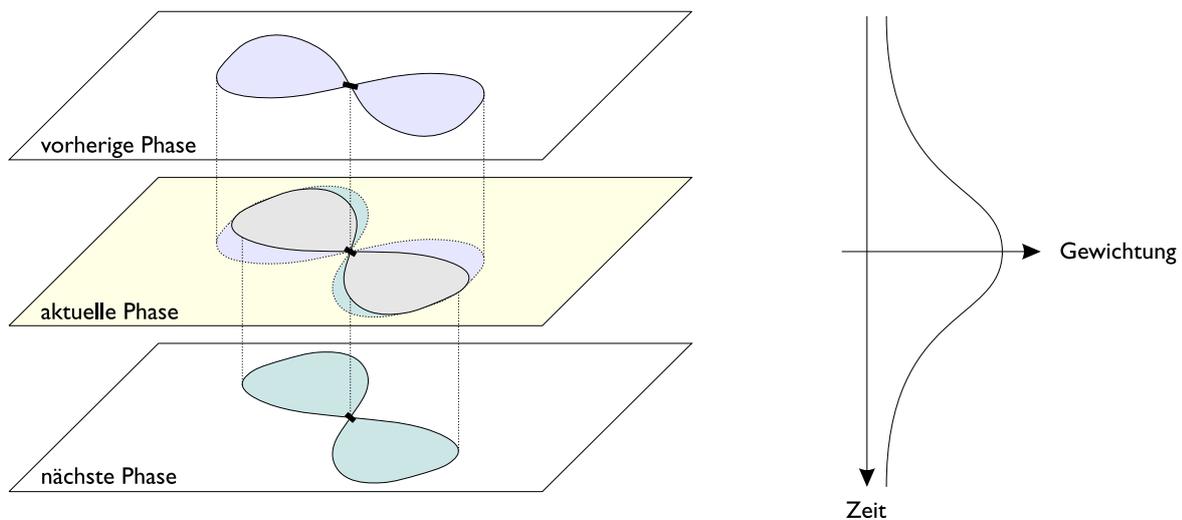


Abbildung 6.1: Votierung über Phasen des Herzzyklus hinweg. Zusätzlich zu den Stimmen aus dem Votierfeld der aktuellen Phase werden an jeder Stelle Stimmen aus den Votierfeldern der benachbarten Phasen gesammelt, gewichtet und aufaddiert. Auf diese Weise werden vorherrschende Strukturen verstärkt und schwächere unterdrückt.

so viele Berechnungen durchgeführt werden wie bei der gewöhnlichen Votierung (z. B. fünfmal so viele Berechnungen bei zwei Nachbarn).

6.2 Krümmung

Die Berücksichtigung der Krümmungsinformation, d. h. die Krümmung oder zumindest ihr Vorzeichen, ermöglicht eine bessere Fortführung von Kurven, so dass z. B. ein unterbrochener Kreis wieder zu einem Kreis vervollständigt wird [TM02] und nicht zu einem ausgebeulten Gebilde (die Enden würden ohne Berücksichtigung der Krümmungsinformation geradlinig fortgesetzt). Die Krümmung bzw. ihr Vorzeichen an einer Stelle wird aus der Nachbarschaft mit Hilfe der bereits bekannten Votierfelder ermittelt.

Abbildung 6.2 zeigt die Erweiterung des Schemas, die durch Berücksichtigung der Krümmungsinformation hinzukommt. Der erste Schritt – das Ermitteln der Krümmungsinformation – tritt neu hinzu, während im zweiten Schritt die Tensorvotierung abgewandelt werden muss, um die zusätzlich gewonnene Information zu berücksichtigen.

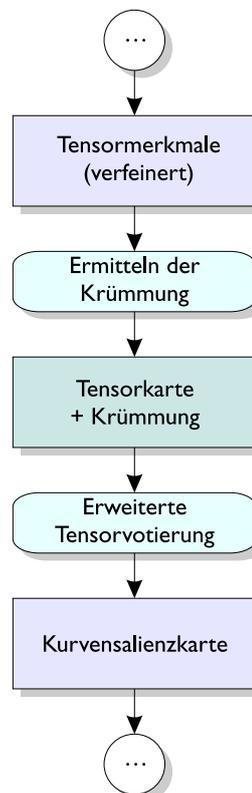


Abbildung 6.2: Tensorvotierung unter Berücksichtigung der Krümmung.

6.2.1 Verfahren nach Tang und Medioni

Nach der ursprünglichen Idee von Tang und Medioni [TM02] wird in dem ersten Schritt mit Hilfe der Tensorvotierung das Vorzeichen der Krümmung jedes Tensors geschätzt und diese anschließend in eine der folgenden Klassen eingeteilt: 1. planar, 2. elliptisch, 3. parabolisch oder 4. hyperbolisch, ein Ausreißer oder eine Diskontinuität. Damit weiß man außer im letzten Fall für jede Stelle, auf welcher Seite sich eine Kurve (bzw. Oberfläche) anschmiegt. Für die nachfolgende Tensorvotierung werden dann abgewandelte Votierfelder verwendet, die diese Kenntnis nutzen,

um bessere Ergebnisse zu erzielen und das Verfahren zu beschleunigen, z. B. berücksichtigt man nur eine Hälfte des grundlegenden Stab-Votierfeldes bei elliptischen Punkten.

Schätzung des Vorzeichens der Krümmungen

Für jeden Eingabe-Tensor O wählt man die Stabkomponente \hat{e}_1 (bezeichnet mit \hat{n}_O) als Bezug für das Vorzeichen die Krümmung; \hat{n}_O steht bekanntlich für die bevorzugte Normalenrichtung des Tensors. Nun betrachtet man die Tensoren P, Q und R in der Nachbarschaft von O (bezeichnet als $N(O)$), die dem Ausmaß des Votierfeldes entspricht. \hat{n}_P, \hat{n}_Q und \hat{n}_R seien die entsprechenden Stabkomponenten von P, Q und R , die durch einen ersten Votierdurchgang oder eine Vorverarbeitung bei Grauwertbildern gewonnen wurden (siehe Abbildung 6.3).

Das Vorzeichen der Krümmung von O wird geschätzt, indem man das Stab-Votierfeld wie gewohnt an O mit dessen Stabkomponente \hat{n}_O ausrichtet und dann die Stimmen aus dem Votierfeld an den Stellen P, Q und R betrachtet. Die Richtungen dieser Stimmen seien mit \hat{n}_P^O, \hat{n}_Q^O und \hat{n}_R^O bezeichnet. O sammelt nun die eigentlichen Stimmen v_P (bzw. v_Q und v_R) für das Vorzeichen der Krümmung ein, deren Richtung $v_P/\|v_P\|_2$ und Stärke $\|v_P\|_2$ mit Hilfe von Tabelle 6.1 ermittelt wird. Diese Art der Votierung unterscheidet sich von der bisherigen (in Abschnitt 2.4.2) dadurch, dass die Stimmen zum einen anders ermittelt werden und zum anderen von den Tensoren in der Nachbarschaft eingesammelt und nicht an sie ausgeteilt werden (ein Fan-In mit dem Votierfeld anstatt eines Fan-Out, siehe Abbildung 2.11).

	P (oberhalb)	Q (gleiche Höhe)	R (unterhalb)
Richtung	$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 1 \end{pmatrix}$	$\frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix}$
Stärke	$\frac{ \hat{n}_P^O \cdot \hat{n}_P }{\ O-P\ _2^2}$	$\frac{ \hat{n}_Q^O \cdot \hat{n}_Q }{\ O-Q\ _2^2}$	$\frac{ \hat{n}_R^O \cdot \hat{n}_R }{\ O-R\ _2^2}$
Vorzeichen	positiv	null	negativ

Tabelle 6.1: Richtung und Stärke einer Stimme bei Votierung für das Vorzeichen der Krümmung.

Klassifizierung

Nachdem alle Stimmen v_P innerhalb der Nachbarschaft von O ermittelt worden sind, muss der Tensor O nun in eine Klasse eingeteilt werden. Dazu wird zuerst der Mittelwert M und die

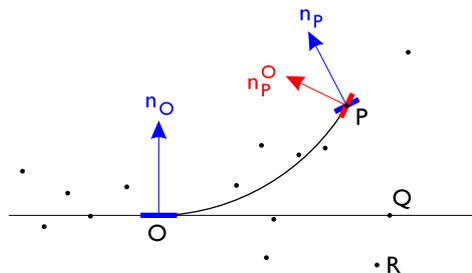


Abbildung 6.3: Votieren für die Krümmung. \hat{n}_O und \hat{n}_P sind die Richtungen der Stabkomponenten von O und P . \hat{n}_P^O ist die Richtung der Stimme in dem an O ausgerichteten Stab-Votierfeld (die bevorzugte Verbindungslinie von O nach P wurde zur Verdeutlichung eingezeichnet).

Kovarianz-Matrix S der Verteilung der Stimmen berechnet:

$$M = (M_x \quad M_y)^T = \frac{1}{n} \sum_{P \in N(O)} v_P \quad (6.4)$$

wobei n die Anzahl der Punkte innerhalb von $N(O)$ angibt. Weiterhin definiert man den Quotienten $\mu = \frac{M_x}{M_y}$ (laut Definition in Tabelle 6.1 gilt $M_y > 0$).

Für $p = 1, 2, \dots, n$ wird $v'_p = v_p - M$ und eine $(2, n)$ -Matrix $B = (v'_1 \ v'_2 \ \dots \ v'_n)$ definiert, d. h. v'_p gibt die Abweichung von v_p vom „gemittelten Vorzeichen“ M an. Mit Hilfe von B berechnet man nun die Kovarianz-Matrix:

$$S = \frac{1}{n-1} B B^T \quad (6.5)$$

S ist eine positiv semidefinite $(2, 2)$ -Matrix. Die Gesamtvarianz von S ergibt sich dann aus der Spur: $\Sigma = \text{spur}(S)$. Mit Hilfe von μ und Σ lässt sich O schließlich folgendermaßen klassifizieren:

- $|\mu| \approx 0, \Sigma \approx 0$. Die Krümmung ist (innerhalb einer geringen Toleranz) Null in der Nachbarschaft $N(O)$. Das bedeutet, O ist lokal *planar*.
- $|\mu| \not\approx 0, \Sigma \approx 0$. Punkte in der Nachbarschaft $N(O)$ befinden sich auf *einer* Seite ($|\mu| \not\approx 0$) und bevorzugen eine *weiche* Verbindung ($\Sigma \approx 0$). Das bedeutet, O ist lokal *elliptisch*.
- $|\mu| \approx 0, \Sigma \not\approx 0$. Die aufsummierten Stimmen heben sich gegenseitig auf ($\Sigma \not\approx 0$). Punkte in der Nachbarschaft $N(O)$ bevorzugen keine bestimmte Seite ($|\mu| \approx 0$). Das bedeutet, O ist lokal *hyperbolisch* oder ein Ausreißer oder eine Diskontinuität.
- $|\mu| \not\approx 0, \Sigma \not\approx 0$. Punkte in der Nachbarschaft $N(O)$ befinden sich auf *einer* Seite ($|\mu| \not\approx 0$), obwohl es Stimmen mit einer Krümmung von Null gibt ($\Sigma \not\approx 0$). Das bedeutet, O ist lokal *parabolisch*.

Tensorvotierung mit speziellen Votierfeldern

Nach Beendigung der Klassifizierung fährt man mit dem zweiten regulären Votierdurchgang fort. Er wird jedoch leicht abgewandelt, indem anstelle des einen, grundlegenden Stab-Votierfeldes drei unterschiedliche Votierfelder zum Einsatz kommen (siehe Abbildung 6.4). Ansonsten ändert sich nichts an der Tensorvotierung. Die Wahl des Votierfeldes für jeden Tensor ergibt sich aus folgenden Fällen:

- *Ebene Punkte*. Es kommt ein abgeflachtes Stab-Votierfeld zum Einsatz, das dem grundlegenden Stab-Votierfeld entspricht, jedoch bei größerer Krümmung stärker abgeschwächt ist.¹
- *Elliptische und parabolische Punkte*. Im grundlegenden Stab-Votierfeld wird lediglich die Hälfte berücksichtigt, die dem geschätzten Vorzeichen der Krümmung entspricht.
- *Hyperbolische Punkte, Ausreißer und Diskontinuitäten*. Bei diesen Punkten gab es uneinheitliche Stimmen bei der Votierung für das Vorzeichen der Krümmung, weshalb das grundlegende Stab-Votierfeld unverändert angewendet wird.

¹Die Konstante c in Gleichung 2.11 wird erhöht.

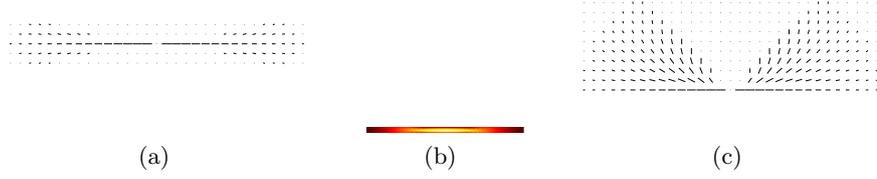


Abbildung 6.4: Spezielle Votierfelder bei Berücksichtigung des Vorzeichens der Krümmung. (a) und (b) zeigen Orientierung und Stärke des Votierfeldes für planare Punkte. (c) zeigt das halbseitig definierte Votierfeld für elliptische Punkte.

6.2.2 Verfahren nach Fischer

Ein anderer Ansatz zur Berücksichtigung der Krümmung nach Fischer [Fis03] verfolgt die Idee, die Krümmungen der bevorzugten Verbindungslinien zweier Merkmale für eine spezielle Krümmungsvotierung zu nutzen. Auf diese Weise erhält man für jeden Tensor eine geschätzte Krümmung und verwendet dann in der Tensorvotierung individuell angepasste Votierfelder. Der Ablauf ist in Abbildung 6.2 dargestellt.

Schätzung der Krümmungen

Zur Schätzung der Krümmung jedes Merkmals wird ein separater Votierdurchgang mit skalaren Werten (Tensoren nullter Ordnung) durchgeführt. Dabei werden die für jede Stelle des grundlegenden Stab-Votierfeldes berechneten Krümmungen aufaddiert (und nicht die Tensoren zweiter Ordnung) und man erhält die geschätzte Krümmung an einer Stelle i :

$$\kappa_i = \frac{1}{|N(i)|} \sum_{k \in N(i)} (\lambda_{1,k} - \lambda_{2,k}) \rho_{k,i} \quad (6.6)$$

wobei $N(i)$ die Nachbarschaft des i -ten Merkmals angibt, d. h. das Ausmaß des Votierfeldes. $\lambda_{1,k}$ und $\lambda_{2,k}$ sind die beiden Eigenwerte des k -ten Merkmals; $\lambda_{1,k} - \lambda_{2,k}$ entspricht der Stabsalienz des k -ten Merkmals. $\rho_{k,i}$ ist die Krümmung der bevorzugten Verbindungslinie zwischen den Merkmalen i und k ; sie ist in Gleichung 2.11 definiert.

Tensorvotierung mit speziellen Votierfeldern

Die Tensorvotierung muss wie bei dem Ansatz nach Tang und Medioni leicht verändert werden, um die Krümmung bei der Votierung zu berücksichtigen. In diesem Ansatz kommen jedoch keine vordefinierten Votierfelder zum Einsatz, sondern es wird für jedes Merkmal ein individuell angepasstes Stab-Votierfeld berechnet. Dabei verlagert sich die räumliche Gewichtung zugunsten der geschätzten Krümmung und es ergibt sich folgende Abschwächungsfunktion (vgl. Gleichung 2.11):

$$DF(s, \rho) = e^{-\left(\frac{s^2 + c(\rho - \kappa)^2}{\sigma_v^2}\right)} \quad (6.7)$$

Die Orientierung der Tensoren im Feld wird vom grundlegenden Stab-Votierfeld ohne Veränderung übernommen (vgl. Gleichung 2.10).

Es ist anzumerken, dass planare Punkte ($\kappa \approx 0$) nicht gesondert durch eine Anpassung des Votierfeldes berücksichtigt werden. Das bedeutet, sie werden wie hyperbolische Punkte, Ausreißer und Diskontinuitäten behandelt und es wird das grundlegende Stab-Votierfeld verwendet.

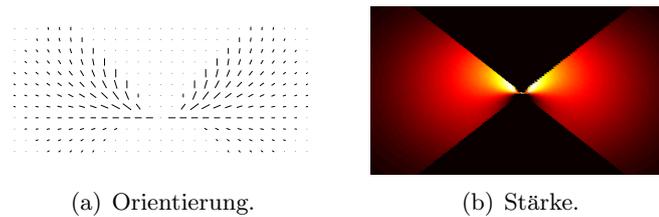


Abbildung 6.5: Spezielles Votierfeld bei Berücksichtigung der Krümmung nach Fischer.

6.3 Polarität

Dieser in [Lee98] ausführlicher dargestellte Aspekt sei nur der Vollständigkeit halber erwähnt, findet jedoch in dieser Arbeit keine weitere Anwendung. Die Polarität erlaubt es, Regionen durch ihre Umrisslinien darzustellen. Dabei bezeichnet sie, welche Seite innen und welche außen liegt. Die normale Tensorvotierung ermittelt nur Kanten und Ecken, kann aber nicht zwischen „Innen“ und „Außen“ unterscheiden, denn die verwendeten Tensoren zweiter Ordnung sind zwar gerichtet, aber nicht polarisiert (siehe Abbildung 6.6).

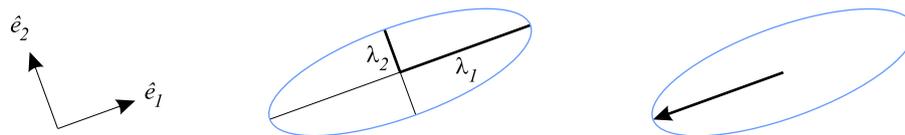


Abbildung 6.6: Veranschaulichung der Polarität. Links ist ein Tensor zweiter Ordnung als Ellipse dargestellt (mit seinem zugehörigen Koordinatensystem); es ist zwar eine Vorzugsrichtung, aber keine bevorzugte Seite zu erkennen. Rechts ist die Polarität eingezeichnet.

Das Tensor Voting Verfahren kann relativ einfach um die Berechnung der Polarität erweitert werden. Abhilfe schafft man durch Hinzunahme von Tensoren erster Ordnung, d. h. Vektoren, die eine Länge von -1 bis 1 haben und die Polarisierung eines Merkmals angeben (die Länge wird als *Polaritätssalienz* bezeichnet). Somit kennt man für jedes Merkmal sowohl einen Tensor zweiter Ordnung, der die Vorzugsrichtung und Stab- bzw. Ballsalienz angibt, als auch einen Tensor erster Ordnung, der die Polarisierung angibt. Bei der anfänglichen Kodierung der Merkmale werden die Polaritäten mit -1, 0 oder 1 initialisiert. Es wird dann eine getrennte Votierung für die Polarität eingeführt, die anstelle der Tensoren zweiter Ordnung mit Vektoren rechnet und eine abgewandelte Form des grundlegenden Stab-Votierfeldes benutzt (Stärke und Vorzugsrichtungen werden als Vektoren und nicht als Matrizen dargestellt). Auf diese Weise erhält man eine Salienzkarte für die Polarität, die als zusätzliche Information für die weitere Auswertung bereitsteht.

6.4 Mehrere Iterationen des Votierdurchgangs

Es sei abschließend noch eine weitere Idee genannt, die Tensorvotierung zu erweitern: mehrere Iterationen des Votierdurchgangs. Das Tensor Voting Verfahren selbst ist nicht iterativ, sondern besteht je nach Eingabe aus einem oder zwei Votierdurchgängen. Folgt man dem Grundschemata, so dient der erste Votierdurchgang (mit der Ballkomponente) der Schätzung der Richtungsinformation und der zweite (mit der Stabkomponente) der Vervollständigung der Merkmale. Bei der Verarbeitung von Grauwertbildern wird der erste Votierdurchgang durch ein Verfahren zur Merkmalsextraktion ersetzt. Nach dem zweiten Votierdurchgang extrahiert man schließlich Kurven und Kreuzungspunkte.

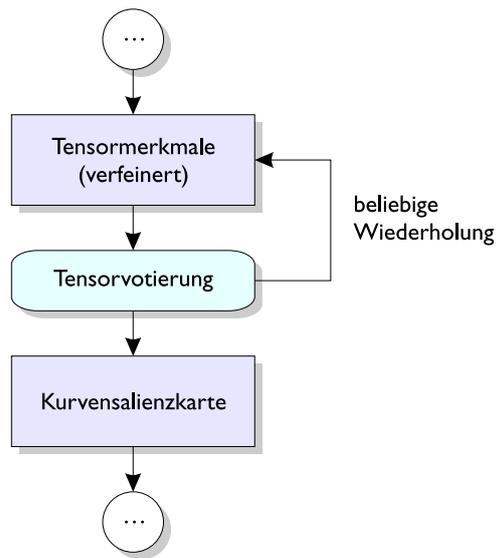


Abbildung 6.7: Mehrere Iterationen der Tensorvotierung.

Eine Idee, die Ergebnisse zu verfeinern und deren Qualität zu verbessern, besteht nun darin, den zweiten Votierdurchgang einmal oder mehrmals zu wiederholen. Dadurch sollen gerichtete Strukturen verstärkt, Lücken überbrückt und Störungen vermindert werden. Allerdings können durch den Ansatz auch unerwünschte Strukturen verstärkt und sogar Verbindungen zwischen Kurvensegmenten hergestellt werden, die nicht zusammengehören. Ob dieser Ansatz sinnvoll ist, bleibt also zu untersuchen.

Ergebnisse und Ausblick

Als Abschluss werden in diesem Kapitel die Ergebnisse der gesamten Arbeit zusammengefasst und ein Ausblick auf Verbesserungen des bestehenden Verfahrens und auf mögliche, zukünftige Erweiterungen gegeben.

7.1 Zusammenfassung

Die vorangegangenen Kapitel haben einen mehrstufigen Bildverarbeitungsprozess vorgestellt, mit dessen Hilfe die Umrisslinien der linken (und rechten) Herzkammer in Magnetresonanztomographie-Aufnahmen des Herzens gefunden werden können. Großes Gewicht fiel dabei auf das noch relativ junge Tensor Voting Verfahren, das in vielerlei Hinsicht vielversprechend erscheint, bisher jedoch noch keine größere Verbreitung gefunden hat. Dessen Anwendung auf Grauwertbilder ist noch neu und kaum untersucht (siehe Babós [Bab03]). Umso interessanter ist deshalb der Versuch, das Verfahren auf ein praktisches Problem anzuwenden. Weil das Tensor Voting Verfahren Bilder auf einer niedrigen abstrakten Stufe verarbeitet, musste eine Methode gefunden werden, dessen Ergebnisse weiterzuverarbeiten und die gesuchte Form der Herzkonturen zu finden. Dazu wurden aufwändige, speziell auf das Problem zugeschnittene Teilschritte entwickelt, die die Anforderungen mehr oder weniger gut erfüllen. In Kapitel 6 wurden auch einige spezielle und allgemeine Methoden zur Erweiterung des Tensor Voting Verfahrens vorgestellt, deren ausführliche Auswertung jedoch noch aussteht.

Während der Arbeit wurde als Exkurs außerdem das Verfahren der Active Shape Models untersucht, das einen komplett anderer Ansatz verfolgt (siehe Anhang A). Ein einfacher Vergleich der beiden Verfahren ist allerdings nicht möglich – nur ein systematischer Vergleich einer ganzen Bandbreite von Ergebnissen aus Herzbildern ist denkbar (mit Standard- und Ausnahmefällen).

Im folgenden sind nun die wichtigsten Ergebnisse der Diplomarbeit in einer Kurzfassung dargestellt:

- ⊕ Das Tensor Voting Verfahren ist nicht iterativ und daher der benötigte Rechenaufwand einfach abschätzbar.
- ⊕ Das Tensor Voting Verfahren benötigt nur einen Parameter, den Skalenfaktor σ_v , dessen Wahl in der Regel unkritisch ist.

- ⊕ Das Tensor Voting Verfahren kann sehr gut mit Rauschen umgehen und selbst bei starkem Rauschen noch Strukturen gruppieren.
- ⊕ Kleine Verbesserungen können durch Erweiterungen des Tensor Voting Verfahrens (aus Kapitel 6) erzielt werden (einige beispielhafte Ergebnisse lassen jedoch keine revolutionären Fortschritte erwarten).
- ⊕ In vielen Fällen liefert die Konturfindung gute Ergebnisse, die nah an die Qualität von Hand gezeichneter Konturen herankommen.
- ⊖ Das Tensor Voting Verfahren kann zwar kleinere Konkavitäten ausgleichen, aber es hat Schwierigkeiten mit den Papillarmuskeln, die im Verhältnis meist sehr groß sind.
- ⊖ Das mehrstufige Verfahren zur Konturfindung unterteilt sich in viele Teilschritte und ist stark an das Problem angepasst, was es sehr kompliziert macht.
- ⊖ Die Konturfindung kann in bestimmten Fällen komplett fehlschlagen (bei ungeeigneter Auswahl von extrahierten Kurvensegmenten).
- ⊖ Die Einstellung vieler Parameter bei der Auswahl und vor allem bei der Gruppierung von Kurvensegmenten ist sehr schwierig und wirkt sich empfindlich auf die Ergebnisse aus (s. u.).
- ⊖ Die Konturfindung bei der rechten Herzkammer ist nur ansatzweise gelöst und liefert nur grobe Ergebnisse.
- ⊖ Die Unterscheidung der inneren und äußeren Kontur der rechten Herzkammer ist mit dem vorgestellten Verfahren nicht möglich. Die Muskelschicht der rechten Herzkammer ist oftmals so dünn, dass das Verfahren nur eine Kante ausmacht.

7.2 Mögliche Verbesserungen

Das Fehlschlagen der Konturfindung lässt sich teilweise dadurch beheben, dass die Parameter bei der Auswahl der Kurvensegmente (Schwellwert T_I der Intensitätsmaske, Position und Nachbarschaft bei der Auswahl der Kurvensegmente für die Epikontur) und des „Gradient Vector Flow“ Verfahrens (Elastizität, Rigidität, Viskosität, externe Kraft und innerer Druck) speziell angepasst werden. Die Parameter sollten in den Phasen des Herzzyklus unterschiedlich gewählt werden, um den unterschiedlichen Anforderungen gerecht zu werden, beispielsweise schrumpft die innere Kontur während der Kontraktion des linken Ventrikels erheblich und ist dadurch stärker gekrümmt als in den übrigen Herzphasen.

Das größte Potential zur Verbesserung liegt sicherlich in der letzten Verarbeitungsstufe, denn die Verfahren zur Kurvenextraktion liefern bereits sehr genaue Ergebnisse. Auch die anschließende Gruppierung der Segmente mit Hilfe des „Gradient Vector Flow“ Verfahrens funktioniert bereits sehr gut, sofern die „richtigen“ Kurvensegmente ausgewählt wurden. Das bedeutet, es müsste hauptsächlich die Auswahl der Kurvensegmente verbessert werden.

Spreeuwers und Breeuwer [SB01] nutzen bei Aufnahmen mit Kontrastmittel die Helligkeitsänderung des Herzmuskels, der während der Kontraktion stärker durchblutet wird und daher mehr Kontrastmittel aufnimmt und heller erscheint. Aus den Unterschieden der Intensitätswerte lässt sich ein Differenzbild berechnen, aus dem man relativ einfach die äußere Kontur der linken Herzkammer ermitteln kann. Die Methode hat Ähnlichkeit mit dem Ansatz des Varianzbildes, setzt allerdings voraus, dass die Aufnahmen mit einem Kontrastmittel gemacht wurden.

7.3 Diskussion

Das Tensor Voting Verfahren ist sehr effektiv bei der Vervollständigung spärlicher Eingabemerkmale, d. h. wenn nur an wenigen Stellen Eingabemerkmale vorhanden sind und die Zwischenräume ausgefüllt werden müssen (siehe Abbildung 2.2). Allerdings bringt es nur kleine Verbesserungen in dem hier vorgestellten mehrstufigen Verfahren bei der Anwendung auf die medizinischen Grauwertbilder. Die Gruppierungsfähigkeit der Tensorvotierung wirkt sich hier schwächer aus als vermutet. Möglicherweise hängt dies mit der Komplexität der Bildstrukturen zusammen: die feinen und oftmals nahe beieinander liegenden Strukturen machen eine kleine Skala (einen kleinen Skalenfaktor) erforderlich, bei der weniger Strukturen gruppiert werden können als bei einer großen Skala. Eine große Skala wiederum ist ungünstig, weil dadurch nur die herausragenden Strukturen erkannt werden, die sowieso relativ einfach zu extrahieren sind, die problematischen Strukturen aber verloren gehen. Abhilfe schaffen eventuell Erweiterungen des Tensor Voting Verfahrens, die die vorherrschende Krümmung berücksichtigen und mit mehreren Skalen arbeiten (siehe Abschnitt 6.2 bzw. 7.4.1).

Viele der erzielten Ergebnisse deuten darauf hin, dass die Verfahren zur „low-level“ Extraktion der Kurvensegmente weitgehend austauschbar sind, d. h. dass dafür sowohl das Tensor Voting Verfahren als auch das Strukturtensor-Verfahren und der Canny-Operator geeignet sind. Zwar liefert das Tensor Voting Verfahren in der Regel bessere Ergebnisse als die anderen beiden Verfahren, aber die Unterschiede sind nicht deutlich genug, um die Qualität der gefundenen Konturen spürbar zu verbessern.

Viel entscheidender für die Qualität der Ergebnisse ist die Weiterverarbeitung der extrahierten Kurvensegmente. Dies liegt daran, dass die Anwendung von Modellwissen einen deutlich größeren Einfluss ausübt als die Genauigkeit und Vollständigkeit der allgemein anwendbaren Verfahren der Merkmalsextraktion. Bei einem Bottom-Up-Ansatz sollte deshalb zuerst eine ausgereifte Methode zur Gruppierung von Kurvensegmenten gefunden werden, bevor man ein höher entwickeltes Verfahren wie die Tensorvotierung in der Vorverarbeitung einsetzt – es sei denn, die Tensorvotierung kann wie oben genannt so verbessert werden, dass sie auch den meisten Teil der Gruppierung leistet und ihre Ergebnisse ohne größere Weiterverarbeitung übernommen werden können.

Aufgrund dieser Ergebnisse stellt sich noch eine andere Frage: ist ein anderer Ansatz für die Konturfindung vielleicht besser geeignet als der hier vorgestellte, z. B. der der Active Shape Models? Lässt man mal die Qualität der Ergebnisse außer Acht (deren Vergleich mit anderen Verfahren noch aussteht), so ergibt sich allein durch die Kompliziertheit des mehrstufigen Verfahrens ein großes, praktisches Problem: es erfordert einen hohen Aufwand bei der Einarbeitung, Wartung und Anpassung an sich ändernde Anforderungen und ist damit fehleranfälliger und kostspieliger als eine einfachere Methode, wie beispielsweise das Verfahren der Active Shape Models. Auch bestimmte Aspekte der Problematik bei den Herzbildern, wie z. B. die Unterscheidung der inneren und äußeren Kontur der rechten Herzkammer, lassen sich vermutlich mit letzterem Verfahren einfacher lösen, da es von Anfang an Modellwissen berücksichtigt.

Diese Frage führt noch zu einer ganz anderen Idee zur Behebung der bestehenden Probleme: die Kombination unterschiedlicher Ansätze, d. h. die Kombination eines statistischen Verfahrens mit einem Verfahren, das geometrische Merkmale extrahiert. Ein solcher Ansatz würde die Stärken beider Seiten vereinen: die Berücksichtigung der Statistik einerseits und die genaue Lokalisation von Merkmalen andererseits. Das Verfahren könnte vorrangig die extrahierten Merkmale heranziehen, in Bereichen mit größerer Unsicherheit allerdings die gesuchte Form gemäß der Statistik vervollständigen. So könnte ein Verfahren entstehen, das sehr robust und dennoch verhältnismäßig einfach ist.

Welches Verfahren letztlich zum Erfolg führt, muss sich allerdings erst noch zeigen. . .

7.4 Ausblick

Die folgenden Abschnitte stellen nun noch zwei Ideen für mögliche, zukünftige Erweiterungen des bisherigen Verfahrens und für neue Ansätze vor.

7.4.1 Multi-scale Tensor Voting

In dieser Arbeit wird immer mit einem festen Skalenfaktor σ_v gearbeitet. Er wird einmal gewählt und an das Problem angepasst, dann aber während des Verfahrens nicht mehr geändert. Dies funktioniert allerdings nur, solange in einem Bild Strukturen von ähnlicher Größe und Detailliertheit vorkommen. Sobald die Strukturen in den genannten Punkten stärker voneinander abweichen, kann die Tensorvotierung nur noch einen Teil von ihnen gut erkennen, während der andere verloren geht (wenn die Strukturen zu fein sind) oder nicht mehr zusammengefasst werden kann (wenn die Strukturen zu grob sind). Medioni [Med02] schlägt zur Behebung dieser Probleme einen mehrstufigen Ansatz vor, sozusagen eine Tensorvotierung mit mehreren Skalen („Multi-scale Tensor Voting“). Der Ansatz steckt noch in der Entwicklung, die Grundidee besteht jedoch darin, die Tensorvotierung mit einer kleinen Skala zu beginnen und Details zu extrahieren und dann mit größeren Skalen zu wiederholen. Die einzelnen Ergebnisse kann man nutzen, um einerseits feine Strukturen extrahieren und andererseits weiter auseinander liegende Strukturen verbinden zu können.

Bei der Anwendung auf die Herzbilder könnte man so die grobe Form der Konturen mit einer großen Skala ermitteln (und dadurch die Papillarmuskeln aussparen) und danach die genaue Position der einzelnen Konturabschnitte mit einer kleinen Skala bestimmen. Die Einzelheiten dieser Methode müssen allerdings erst noch erarbeitet und die Auswirkungen auf die Merkmalsextraktion untersucht werden.

7.4.2 Dreidimensionale Rekonstruktion des Herzens

Das Tensor Voting Verfahren ist nicht auf zwei Dimensionen beschränkt und lässt sich prinzipiell auf beliebig viele Dimensionen erweitern. Denkbar wäre deshalb eine dreidimensionale Rekonstruktion der Herzkammern und Herzmuskulatur. Für diese anspruchsvolle Aufgabe müsste allerdings jede Verarbeitungsstufe erweitert und angepasst werden. So müsste man bei der Eingabe mehrere Schichtaufnahmen parallel betrachten, d. h. dreidimensionale Darstellungen mit Volumenelementen (Voxel) anstelle der zweidimensionalen Bilder mit Pixeln verwenden. Die Erweiterung auf die dritte Dimension würde dreidimensionale Votierfelder und Tensorkarten erforderlich machen, die Tensoren selbst würden formal als $(3, 3)$ -Matrizen und bildlich als Ellipsoide dargestellt werden und man bräuchte raffiniertere Methoden, Oberflächen, Kurven und Kreuzungspunkte aus den Salienzkarten zu extrahieren (siehe [MLT00, Lee98, Tan00]). Desweiteren müsste der Strukturtensor auf drei Dimensionen erweitert werden oder durch ein anderes Verfahren ersetzt werden, das aus der Grauwertdarstellung der Volumenelemente Eingabemerkmale für die Tensorvotierung gewinnt. Und schließlich müsste die Weiterverarbeitung der extrahierten Oberflächen- und Kurvensegmente angepasst werden.

Ein anderer, weniger aufwändiger Ansatz zur dreidimensionalen Rekonstruktion wäre sicherlich, die Schichtaufnahmen des Herzens mit dem bekannten Verfahren zu untersuchen und dann die einzelnen Ergebnisse zu einer dreidimensionalen Struktur zusammenzufassen. Dabei ginge allerdings der Vorteil verloren, dass alle drei Dimensionen gleichzeitig betrachtet werden und diese sich gegenseitig beeinflussen und ergänzen können. Eine Erweiterung auf drei Dimensionen ist letztlich jedoch nur anzuraten, wenn die bestehenden Probleme im zweidimensionalen Fall gelöst sind. \square

Active Shape Models

Dieses Kapitel stellt einen kurzen Ausflug zu einem Verfahren mit einem völlig anderen Ansatz als das mehrstufige Tensor Voting Verfahren dar – dem Verfahren der Active Shape Models (ASM). Dieses von Cootes u. a. [Coo00, CT01] entwickelte Verfahren beruht auf der Erstellung eines statistischen Modells anhand von Trainingsdaten, wobei die für die gesuchte Form charakteristischen Veränderungen ermittelt werden. Bei der Anwendung auf ein beliebiges Eingabebild geht man von der statistisch gemittelten Form aus und passt diese schrittweise den Gegebenheiten des Bildes, d. h. den Intensitätswerten, an. An dieser Vorgehensweise erkennt man, dass es sich hierbei um einen Top-Down-Ansatz handelt.

Die folgenden Abschnitte legen die Grundlagen der Active Shape Models dar, wegen der Kürze sei jedoch für weitere Details auf die oben genannte, weiterführende Literatur verwiesen. Abschnitt A.1 geht auf die Erstellung des statistischen Modells ein, Abschnitt A.2 auf die Anpassung des Modells an ein Eingabebild und der letzte Abschnitt schließlich auf Ergebnisse.

A.1 Erzeugung des statistischen Modells

Für die Erzeugung des statistischen Modells wird ein Trainingsdatensatz benötigt, der das Modellwissen in Form von annotierten Grauwert-Bildern widerspiegelt, d. h. Originalbilder und die jeweils zugehörige, gesuchte Form. Als *Form* wird die gesamte geometrische Information ausschließlich des Ortes, der Größe und der Orientierung bezeichnet. Abbildung A.1 zeigt ein Beispiel für eine Form der linken Herzkammer. Das am Ende erzeugte Modell stellt in kompakter Weise die möglichen Veränderungen der gesuchten Form in einem Bild dar, lässt jedoch keine willkürlichen Veränderungen zu.

A.1.1 Auswahl von Landmarken

In jeder Form der Trainingsdaten müssen geeignete Punkte, sogenannte *Landmarken*, ausgewählt werden. Gute Landmarken sind beispielsweise Eck- oder Kreuzungspunkte, jedoch gibt es davon oftmals zu wenige, so dass zwischen ihnen Punkte in regelmäßigen Abständen ausgewählt werden. Wichtig bei der Erzeugung der Trainingsdaten ist, dass als Landmarken in jedem Bild dieselben Strukturen ausgewählt werden, damit die Gemeinsamkeit der Form in den Bildern erfasst werden kann. Man erhält schließlich für eine Form eine Folge von n Punkten $(u_1, v_1), (u_2, v_2), \dots, (u_n, v_n)$.

Daraus bildet man den $2n$ -dimensionalen Vektor x mit

$$x = (u_1, u_2, \dots, u_n, v_1, v_2, \dots, v_n)^T \quad (\text{A.1})$$

Auf diese Weise erhält man für den gesamten Trainingsdatensatz m solcher Vektoren x_i (wenn der Datensatz m Bilder enthält).

A.1.2 Ausrichten der Formen

Da die Modellform unabhängig von Ort, Größe und Orientierung erfasst werden soll, müssen die Formen der Trainingsdaten in einem gemeinsamen Koordinatensystem ausgerichtet und aneinander angepasst werden, bevor man deren Variationen ermitteln kann. Dies kann man z. B. mit der sogenannten Prokrustes-Transformation¹ („Procrustes analysis“) bewerkstelligen, die in [SG02] ausführlicher erläutert wird. Die Idee besteht darin, jede Form so zu verschieben, zu skalieren oder zu drehen, dass die Summe der quadratischen Abstände aller Formen zum Mittel ($D = \sum_{i=1}^m \|x_i - \bar{x}\|_2^2$) minimal wird. Aus diesem Verarbeitungsschritt gehen schließlich die ausgerichteten Formen sowie eine *Referenzform* hervor. Die Referenzform r entspricht dem Mittel aller ausgerichteten Formen, wurde allerdings auf deren ursprüngliche Größe ausgedehnt: $r = \bar{s}\bar{x}$, wobei \bar{s} der mittleren Größe aller Formen vor der Ausrichtung und \bar{x} dem Mittel aller ausgerichteten Formen entspricht. Die Referenzform dient bei der späteren Anwendung des Modells zur Initialisierung.

A.1.3 Modellierung der Variation

Nachdem man die Menge der Formen ausgerichtet hat, kann man deren Variation modellieren. Normalerweise korrelieren Punkte mehrerer Formen miteinander (sonst gäbe es keine Variation oder die Punkte wären zufällig verteilt; dann wären es aber keine Landmarken mehr), was man ausnutzen kann, um zu einer kompakteren Darstellung zu gelangen, d. h. die ursprünglich $2n$ Freiheitsgrade können durch die Abhängigkeiten der Punkte untereinander stark verringert werden. Ein effektiver Ansatz dafür ist die Hauptkomponentenanalyse („Principal Component Analysis“, PCA), die aus der Statistik bekannt ist. Dazu berechnet man zuerst die Kovarianz-Matrix

$$S = \frac{1}{m-1} \sum_{i=1}^m (x_i - \bar{x})(x_i - \bar{x})^T \quad (\text{A.2})$$

und zerlegt diese anschließend in ihr Eigensystem, d. h. man berechnet die Eigenvektoren \hat{e}_i und die zugehörigen Eigenwerte λ_i , wobei $\lambda_i \geq \lambda_{i+1}$. Jeder Eigenvektor gibt die Varianz der Daten in Richtung des entsprechenden Eigenvektors an. Zur Verringerung der Dimensionalität berechnet man dann zuerst die gesamte Varianz $V_T = \sum_{i=1}^{2n} \lambda_i$ und wählt anschließend die t größten Eigenwerte aus, so dass gilt:

$$\sum_{i=1}^t \lambda_i \geq f_v V_T \quad (\text{A.3})$$

wobei f_v den Anteil der Variation angibt, den man beschreiben können will, z. B. setzt man $f_v = 0.95$, wenn man 95 % der Variation in den Trainingsdaten beschreiben will.

¹Prokrustes war ein riesiger Unhold der griechischen Mythologie, der Wanderern auflauerte und sie durch Abschneiden der Beine oder Streckung einem Bett anpasste. Er wurde von Theseus überlistet und auf dieselbe Weise getötet.

Wenn man eine Hauptkomponentenanalyse der Daten durchgeführt hat, kann man jeden Vektor x folgendermaßen annähern:

$$x \approx \bar{x} + Pb \quad (\text{A.4})$$

wobei $P = \begin{pmatrix} \hat{e}_1 & \hat{e}_2 & \dots & \hat{e}_t \end{pmatrix}$ die ersten t Eigenvektoren der Kovarianz-Matrix enthält und b ein t -dimensionaler Vektor ist, gegeben durch:

$$b = P^T(x - \bar{x}) \quad (\text{A.5})$$

b definiert die Parameter, die nach Gleichung A.4 die Veränderung der Modellform bewirken (die b_i werden als *Modellform-Parameter* bezeichnet). Die Varianz des i -ten Parameters b_i in Bezug auf die Trainingsdaten ist durch den Eigenwert λ_i gegeben. Damit die Form bei der Anpassung nicht zu stark von den vorgegebenen Formen abweicht, beschränkt man die Parameter b_i auf einen Bereich von $\pm 3\sqrt{\lambda_i}$.² Abbildung A.1 zeigt eine Referenzform der linken Herzkammer und deren Variationen für die ersten drei Modellform-Parameter.

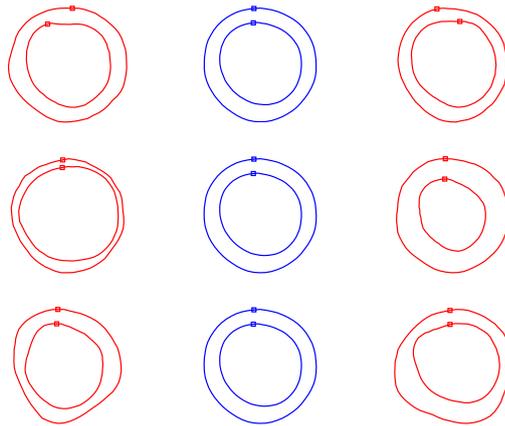


Abbildung A.1: Änderung der Modellform-Parameter. Gezeigt sind die Auswirkungen bei Veränderung der ersten drei Modellform-Parameter (von oben nach unten) um das ± 3 -fache ihrer Standardabweichungen. In der mittleren Spalte ist jeweils die Referenzform zu sehen. In jeder Form sind die Anfangspunkte der inneren und äußeren Kontur als Referenzen markiert.

A.2 Anpassung der Modellform

Wenn man das statistische Modell erzeugt hat, kann man durch Wahl der Modellform-Parameter eine neue Form erzeugen. Diese muss man jedoch vor einer weiteren Verarbeitung aus dem Koordinatensystem des Modells in das des Bildes transformieren, was man durch Festlegung einer Position (X_t, Y_t) , einer Orientierung θ und einer Skalierung s erreicht. Die Koordinaten der Modellpunkte im Bild ergeben sich dann folgendermaßen:

$$X = T_{X_t, Y_t, \theta, s}(\bar{x} + Pb) \quad (\text{A.6})$$

²Nach der aus der Stochastik bekannten 3σ -Regel bei Normalverteilungen [Fis89]. Ausgefeiltere Methoden verwenden individuelle Bereiche für die Parameter.

wobei $T_{X_t, Y_t, \theta, s}$ eine zusammengesetzte Transformation ist, die eine Rotation um θ , eine Skalierung um s und eine anschließende Translation um (X_t, Y_t) durchführt:

$$T_{X_t, Y_t, \theta, s}(x, y) = \begin{pmatrix} X_t \\ Y_t \end{pmatrix} + \begin{pmatrix} s \cos \theta & -s \sin \theta \\ s \sin \theta & s \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (\text{A.7})$$

Angenommen man hat nun in einem Eingabebild eine Reihe von Punkten Y ausfindig gemacht (mehr dazu in Abschnitt A.2.1), so muss man die Punkte X der Modellform an sie anpassen. Unter Verwendung des quadratischen Abstands führt dies zu der Minimierungsaufgabe

$$\|Y - T_{X_t, Y_t, \theta, s}(\bar{x} + Pb)\|_2^2 \rightarrow \min. \quad (\text{A.8})$$

Das Problem kann durch folgenden iterativen Ansatz gelöst werden, der gewöhnlich innerhalb weniger Iterationen konvergiert:

1. Initialisierung der Modellpunkte x und Modellform-Parameter b (z. B. $b = 0$, so dass man den Mittelwert \bar{x} der Modellform erhält, d. h. $x = \bar{x}$ nach Gleichung A.4)
2. Ermitteln der Transformation, die die Modellpunkte x am besten auf die gefundenen Punkte Y abbildet. Durch eine Prokrustes-Transformation erhält man die Parameter X_t, Y_t, s und θ .
3. Rücktransformation der gefundenen Punkte Y in das Koordinatensystem der Modellform:
 $y = T_{X_t, Y_t, \theta, s}^{-1}(Y)$
4. Projektion von y in die Tangentenebene von \bar{x} (siehe [SG02]): $y' = \frac{y}{y \cdot \bar{x}}$
5. Anpassung der Modellform-Parameter an y' : $b = P^T(y' - \bar{x})$
6. Beschränkung der Parameter auf den festgelegten Bereich, so dass z. B. gilt: $|b_i| < 3\sqrt{\lambda_i}$
7. Berechnung der neuen Positionen der Modellpunkte: $x_{\text{neu}} = \bar{x} + Pb$
8. Setze $x = x_{\text{neu}}$ und gehe zu 2., falls sich die Punkte mehr als um einen geringfügigen Wert ϵ geändert haben, d. h. falls gilt: $\|x - x_{\text{neu}}\|_2^2 > \epsilon$

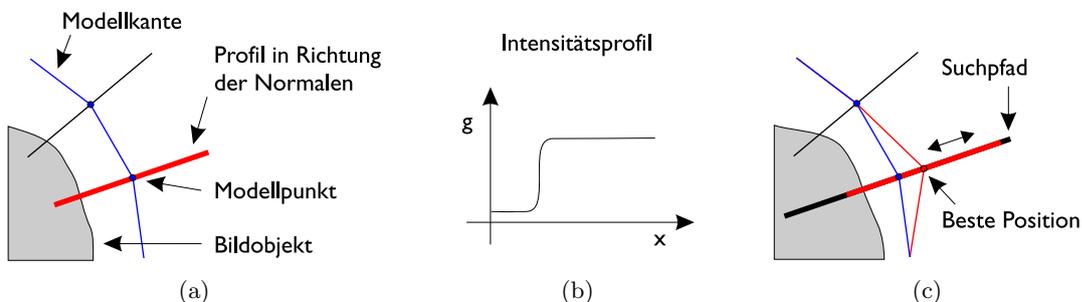


Abbildung A.2: Ermitteln eines Intensitätsprofils und Suche nach einer neuen Position. In Richtung der Normalen werden auf beiden Seiten eines Modellpunktes (a) die Intensitätswerte des Bildes ausgelesen und als Profil dargestellt (b). Bei der Suche nach einer neuen Position für einen Punkt wird entlang eines verlängertes Profils gesucht und die Position mit der besten Übereinstimmung ausgewählt (c).

A.2.1 Ermitteln der neuen Punkte

Nachdem der letzte Abschnitt erläutert hat, wie man die Modellform an eine Menge im Bild gefundener Punkte anpassen kann, bleibt zu klären, wie man die Position dieser Punkte findet. Dazu gibt es unterschiedliche Möglichkeiten, jedoch laufen alle Ansätze auf die Optimierung einer Kostenfunktion hinaus. Die Idee von Cootes besteht darin, an jedem Modellpunkt das Profil entlang der Normalenrichtung zu betrachten (siehe Abbildung A.2(a) und (b)). Man liest aus dem Profil auf jeder Seite eines Punktes k Pixel aus und erhält so insgesamt $2k + 1$ Abtastungen. Anstatt der Intensitätswerte selbst verwendet man deren Ableitung, um Veränderungen der Grauwerte und nicht deren absoluten Betrag zu erfassen (der von Bild zu Bild schwanken kann). Diese abgeleiteten Werte schreibt man für jedes Bild i aus dem Trainingsdatensatz in einen Vektor g_i und normalisiert diesen anschließend:

$$g_i \mapsto \frac{1}{\sum_{j=1}^{2k+1} |g_{i,j}|} g_i \quad (\text{A.9})$$

Es wird angenommen, dass die Menge der Profile aus allen Bildern $\{g_1, g_2, \dots, g_m\}$ normalverteilt ist, und man ermittelt deren Mittelwert \bar{g} und Kovarianz S_g . Damit erhält man ein statistisches Modell für die Grauwertverteilung des Profils an diesem Punkt. Diese Schritte wiederholt man für alle übrigen Punkte.

Bei der Suche nach einer neuen Position für einen Punkt tastet man m Punkte auf dessen beiden Seiten ab, wobei gilt: $m > k$. In diesem langen Profil untersucht man alle $2(m - k) + 1$ möglichen Positionen und wählt diejenige aus, die am besten zum statistischen Modell passt (siehe Abbildung A.2(c)). Als Qualitätsmaß für die Übereinstimmung eines neuen Profils g_s mit dem Modell verwendet man die *Mahalanobis-Distanz*:

$$f(g_s) = (g_s - \bar{g})^T S_g^{-1} (g_s - \bar{g}) \quad (\text{A.10})$$

Eine Minimierung der Mahalanobis-Distanz entspricht dabei einer Maximierung der Wahrscheinlichkeit, dass g_s aus der Verteilung stammt.

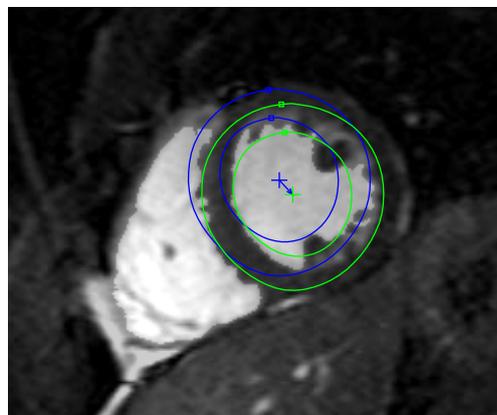


Abbildung A.3: Initialisierung mit der Referenzform. Veranschaulicht ist die Verschiebung der Referenzform (blau) zur Startposition (grün) für das Verfahren; die Bereiche der Intensitätsmaske wurden leicht hervorgehoben.

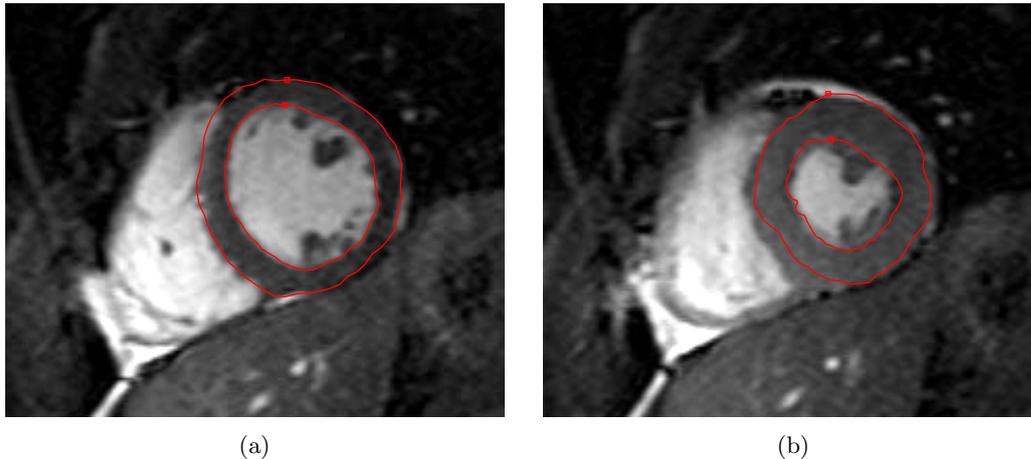


Abbildung A.4: Ergebnisse für das Active Shape Model. (a) und (b) zeigen die durch das ASM ermittelten Konturen der linken Herzkammer in zwei verschiedenen Phasen des Herzzyklus. Die Konturen in (b) wurden nur unzureichend erkannt. Das Verfahren wurde auf ein Bild angewendet, das nicht aus dem Trainingsdatensatz stammt.

A.3 Ergebnisse

Die Initialisierung des Verfahrens spielt eine wichtige Rolle und wurde bei der Anwendung auf die Herzbilder folgendermaßen umgesetzt: anhand der in Abschnitt 5.6.1 beschriebenen Intensitätsmaske wird der Schwerpunkt der linken Herzkammer bestimmt und dann die Referenzform mit ihrem Schwerpunkt dorthin verschoben (veranschaulicht in Abbildung A.3). Dies liefert bereits eine gute Startposition für das Verfahren und führt für gewöhnlich auch zu einer Konvergenz. Abbildung A.4 zeigt zwei Ergebnisse für die Konturdetektion an der linken Herzkammer. Zu sehen ist eine relativ gute (a) und eine schlechte Anpassung (b). Eine schlechte Anpassung kann beispielsweise dadurch entstehen, dass der Trainingsdatensatz keine ausreichende Bandbreite an Beispielen enthält.

Das Verfahren der Active Shape Models hat sich bei der Anwendung auf einige Beispielbilder als sehr vielversprechend erwiesen, jedoch hat auch dieses Verfahren seine Nachteile und es gibt noch viele Aspekte, die (außerhalb des Rahmens dieser Arbeit) zu untersuchen sind – einschließlich eines Vergleichs mit dem verwandten Verfahren der Active Appearance Models, das zusätzlich zur Form von Strukturen deren Textureigenschaften berücksichtigt. Abschließend für dieses Kapitel werden einige Vor- und Nachteile des Verfahrens aufgeführt:

- ⊕ Active Shape Models sind sehr flexibel und können leicht an sich ändernde Anforderungen angepasst werden. In der Regel müssen lediglich die Trainingsdaten angepasst und ein neues statistisches Modell erzeugt werden, was einfacher und schneller geht, als die Implementierung einer höheren Verarbeitungsstufe (siehe Kapitel 5) bei einem Bottom-Up-Ansatz anzupassen. Ein Beispiel hierfür wäre die Hinzunahme weiterer Strukturen wie der rechten Herzkammer.
- ⊕ Das Verfahren umgeht viele Probleme aus Abschnitt 5.2 dadurch, dass ein statistisches Modell aus Originalbildern und vorgegebenen Konturen „erlernt“ wird.
- ⊖ Die Erstellung des Trainingsdatensatzes ist meistens sehr aufwändig, da in vielen Bildern die Landmarken von Hand gesetzt werden müssen.

- ⊖ Die Qualität der Ergebnisse hängt entscheidend von den Trainingsdaten ab, die aus diesem Grund sorgfältig ausgewählt werden sollten.

Hinweise zur Implementierung

Dieses Kapitel weist in aller Kürze auf einige Implementierungsaspekte der vorgestellten Verfahren hin. Umgesetzt wurden diese in Matlab, das sich sehr gut eignet, um ohne großen, zusätzlichen Aufwand prototypische Anwendungen zu entwickeln, Algorithmen auszutesten und zu visualisieren.

B.1 Bilineare Interpolation

Da einige Methoden mit reellwertigen Koordinaten arbeiten, während das zugehörige Bild oder die zugehörige Tensorkarte diskret ist und ganzzahlige Koordinaten hat, müssen Zwischenwerte aus den umliegenden, gegebenen Werten berechnet werden. Wie in Abbildung B.1 zu sehen ist, ermittelt die *bilineare Interpolation* [RK82] den Zwischenwert an Position (x, y) aus den vier angrenzenden Werten in der Matrix G . Es gilt:

$$\begin{aligned}
 G_{\text{int}}(x, y) &= (1 - \Delta x)(1 - \Delta y) G(x_0, y_0) \\
 &\quad + (1 - \Delta x)\Delta y G(x_0, y_0 + 1) \\
 &\quad + \Delta x(1 - \Delta y) G(x_0 + 1, y_0) \\
 &\quad + \Delta x\Delta y G(x_0 + 1, y_0 + 1)
 \end{aligned}
 \tag{B.1}$$

mit

$$x_0 = \lfloor x \rfloor, \quad y_0 = \lfloor y \rfloor, \quad \Delta x = x - x_0, \quad \Delta y = y - y_0.$$

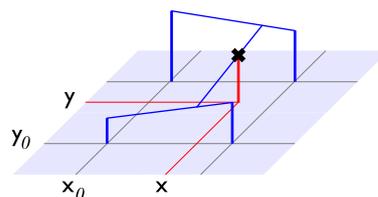


Abbildung B.1: Bilineare Interpolation.

Prozess	Parameter	Wert	Bedeutung
Varianzmaske	T_H	30000	Histogramm-Schwellwert, entspricht etwa der Größe der Maske (in Pixel)
	T_v	berechnet	Schwellwert zur Ermittlung der Maske
	N	7	Größe der Maske ($N \times N$) für morphologischen Operator
	N_{\min}	20	Unterer Schwellwert für morphologischen Operator
	N_{\max}	24	Oberer Schwellwert für morphologischen Operator
Intensitätsmaske	T_I	0.4	Schwellwert zur Unterteilung der Helligkeitsbereiche
Strukturtensor	σ_g	0.5	Standardabweichung des Gauß'schen Tiefpassfilters
	T_e	0.01	Schwellwert für Eigenwerte zur Verringerung der Anzahl der Eingabemerkmale
Tensorvotierung	σ_v	6	Skalenfaktor der Votierfelder
	w	2	Faktor für zeitliche Gewichtung (bei Erweiterung)
	T_{cmap}	0.005	Schwellwert für Kurvensalienzkarte
Kurvenextraktion	L_{\min}	10	Mindestlänge von extrahierten Kontursegmenten
Segmentauswahl	N	5	Größe der zu betrachtenden Nachbarschaft ($N \times N$)
	L_{\min}	5	Mindestlänge von ausgewählten Kurvensegmenten (kleiner als bei der Kurvenextraktion, da die Segmente weiter unterteilt werden können)
Aktive Konturanpassung	α	35	Elastizität
	β	10^4	Rigidität
	γ	1	Viskosität
	κ	2	Externe Kraft
	π	1.33	Innerer Druck

Tabelle B.1: Die Parameter des Extraktionsverfahrens und ihre Bedeutung

B.2 Parameter des Extraktionsverfahrens

B.3 Externe Matlab-Funktionen

Für die Extraktion von Liniestücken aus binären Bildern (z. B. aus einer Kurvensalienzkarte, auf die ein Schwellwert angewendet wurde) kommen einige Funktionen von Kovesi [Kov01] zur Anwendung.

Da die Originalbilder als MPEG-Video-Dateien vorliegen, wird die Funktion `mpgread` [Fot99] zum Auslesen benötigt. Für Bilder im DICOM-Format¹ bietet Matlab bereits eine rudimentäre Funktionalität an.

Die aktive Konturanpassung aus Abschnitt 5.7.1 wurde mit Hilfe der Funktionen für GVF-Snakes aus [XP99] realisiert.

¹Digital Imaging and Communications in Medicine, ein Standard der National Electrical Manufacturers Association (NEMA) für medizinische Bilder.

Die beiliegende CD

Der Diplomarbeit liegt eine CD bei, die zusätzliches Material enthält, darunter eine elektronische Fassung der Diplomarbeit, die Eingabe- und Ergebnisbilder aller Versuche, den Quellcode der Matlab-Funktionen und einiges mehr.

Die folgende Abbildung zeigt die Struktur und den Inhalt der CD:

docs/	PDF-/PS-Version der Diplomarbeit
experiments/	Versuchsergebnisse
bwnoise/	Rauschen bei gewöhnlichem Tensor Voting
concav/	Konkavitäten
diagrams/	Diagramme mit Fehlermaßen
dist/	Unterschiedlich entfernte Strukturen
docs/	PDF-Dateien mit Bildübersicht
graynoise/	Rauschen bei Grauwertbildern
lowcontrast/	Schwache Kontraste
orient/	Diskretisierungseffekte bei Rotation
rampedge/	Rampenförmige Kanten
size/	Größe und Skalenfaktor
packages/	Zusätzliche Software und Matlab-Funktionen
src/	Quellcode der Matlab-Funktionen
README.txt	Weitere Details und Anweisungen

Abbildung C.1: Die Verzeichnisstruktur der beiliegenden CD.

Die Datei `README.txt` enthält weitere Details, Anweisungen zur Benutzung und Hinweise zum Urheberrecht und sollte unbedingt als erstes gelesen werden.

C Die beiliegende CD

- [Bab03] BABÓS, Martin: *Anwendung des Tensor-Voting-Verfahrens auf Grauwertbilder*, Universität Hamburg, Diplomarbeit, Februar 2003
- [Bea78] BEAUDET, P. R.: Rotationally Invariant Image Operators. In: *Proc. International Joint Conference on Pattern Recognition*. Kyoto, Japan, November 1978, S. 579–583
- [Bär01] BÄR, Christian: *Elementare Differentialgeometrie*. 1. Auflage. Berlin, New York : de Gruyter, Januar 2001. – 281 S. – ISBN 3–11–015519–2
- [Can83] CANNY, John F.: Finding Edges and Lines in Images / Massachusetts Institute of Technology. 1983 (AI-TR-720). – Forschungsbericht. – 146 S
- [Coo00] COOTES, Timothy F.: Model-Based Methods in Analysis of Biomedical Images. In: BALDOCK, Richard (Hrsg.) ; GRAHAM, Jim (Hrsg.): *Image Processing and Analysis: A Practical Approach*. 1. Auflage. Oxford University Press, März 2000 (The Practical Approach Series). – ISBN 0–19–963700–8, Kapitel 7, S. 223–248
- [CT01] COOTES, Timothy F. ; TAYLOR, Christopher J.: Statistical Models of Appearance for Computer Vision / University of Manchester. Manchester, U.K., Oktober 2001. – Forschungsbericht
- [Dau88] DAUGMAN, John G.: Complete Discrete 2D Gabor Transforms by Neural Networks for Image Analysis and Compression. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 36 (1988), Nr. 7, S. 1169–1179
- [DN82] DRESCHLER, Leonie ; NAGEL, Hans-Hellmut: On the Selection of Critical Points and Local Curvature Extrema of Region Boundaries for Interframe Matching. In: *Proc. International Conference on Pattern Recognition*, 1982, S. 542–544
- [EZ96] ELDER, James H. ; ZUCKER, Steven W.: Computing Contour Closure. In: BUXTON, Bernard F. (Hrsg.) ; CIPOLLA, Roberto (Hrsg.): *Computer Vision - ECCV '96, Vol. I*. Cambridge, U.K. : Springer-Verlag, 1996 (Lecture Notes in Computer Science Vol. 1064). – ISBN 3–540–61122–3, S. 399–412
- [FDFH95] FOLEY, James D. ; VAN DAM, Andries ; FEINER, Steven K. ; HUGHES, John F.: *Computer Graphics: Principles and Practice in C*. 2. Auflage. Reading, Mass. : Addison Wesley Professional, August 1995. – 1200 S. – ISBN 0–201–84840–6

Literaturverzeichnis

- [FH98] FUCHS, Claudia ; HEUEL, Stefan: Feature Extraction. In: FÖRSTNER, Wolfgang (Hrsg.): *Third Course in Digital Photogrammetry*. Bonn, Februar 1998
- [Fis89] FISZ, Marek: *Wahrscheinlichkeitsrechnung und mathematische Statistik*. 11. Auflage. Berlin : Deutscher Verlag der Wissenschaften, 1989. – 777 S. – ISBN 3-326-00079-0
- [Fis03] FISCHER, Sylvain: *Tensor Voting*. August 2003. – Instituto de Óptica, CSIC, Madrid
- [Fot99] FOTI, David: *mpgread: MPEG to MATLAB Movie Converter*. URL: <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=308>. 1999. – The MathWorks, Inc.
- [För94] FÖRSTNER, Wolfgang: A Framework for Low-Level Feature Extraction. In: EKLUNDH, Jan-Olof (Hrsg.): *Computer Vision - ECCV '94, Vol. II*. Berlin, Heidelberg : Springer-Verlag, 1994 (Lecture Notes in Computer Science Vol. 801). – ISBN 3-540-57957-5, S. 383-394
- [Gáb46] GÁBOR, Dennis: Theory of Communication. In: *Journal of the Institution of Electrical Engineers* 93 (1946), Nr. 26, S. 429-457
- [GM96] GUY, Gideon ; MEDIONI, Gérard: Inferring Global Perceptual Contours from Local Features. In: *International Journal of Computer Vision* 20 (1996), Nr. 1/2, S. 113-133
- [GW01] GONZALES, Rafael C. ; WOODS, Richard E.: *Digital Image Processing*. 2. Auflage. Prentice Hall, November 2001. – 728 S. – ISBN 0-201-18075-8
- [IK88] ILLINGWORTH, John ; KITTLER, Josef: A Survey of the Hough Transform. In: *Computer Vision, Graphics, and Image Processing* 44 (1988), April, S. 87-116
- [Joh01] JOHANSSON, Björn: Multiscale Curvature Detection in Computer Vision / Dept. EE, Linköping University. SE-581 83 Linköping, Sweden, März 2001 (LiU-Tek-Lic-2001:14). – Lic. Thesis. Thesis No. 877, ISBN 91-7219-999-7
- [JP87] JONES, Judson P. ; PALMER, Larry A.: An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. In: *Journal of Neurophysiology* 58 (1987), Nr. 6, S. 1233-1258
- [Kov01] KOVESI, Peter. *MATLAB Functions for Computer Vision and Image Analysis*. URL: <http://www.cs.uwa.edu.au/~pk/Research/MatlabFns/index.html>. 2001
- [KWT88] KASS, Michael ; WITKIN, Andrew ; TERZOPOULOS, Demetri: Snakes: Active Contour Models. In: *International Journal of Computer Vision* 1 (1988), Januar, Nr. 4, S. 312-331
- [Lee98] LEE, Mi-Suen: *Tensor Voting for Salient Feature Inference in Computer Vision*, University of Southern California, Diss., August 1998
- [Lin94] LINDBERG, Tony: Scale-space Theory: A Basic Tool for Analysing Structures at Different Scales. In: *Journal of Applied Statistics* 21 (1994), Nr. 2, S. 224-270. – Supplement on Advances in Applied Statistics: Statistics and Images: 2
- [Lip96] LIPPMANN, Horst: *Angewandte Tensorrechnung*. 2. Auflage. Berlin, Heidelberg : Springer-Verlag, September 1996. – 266 S. – ISBN 3-540-61575-X
- [Lyn00] LYNCH, Patrick J. *Yale University School of Medicine: Cardiothoracic Imaging*. URL: <http://info.med.yale.edu/intmed/cardio/imaging/contents.html>. 2000

- [Med02] MEDIONI, Gérard. *Tensor Voting - Recent Results and Perspectives*. URL: <http://forum.stanford.edu/events/archive/affiliates-week/affiliates-week-slides.html>. März 2002
- [MLT00] MEDIONI, Gérard ; LEE, Mi-Suen ; TANG, Chi-Keung: *A Computational Framework for Segmentation and Grouping*. 1. Auflage. Amsterdam : Elsevier, 2000. – 260 S. – ISBN 0-444-50353-6
- [Neu02a] NEUMANN, Heiko. *Computer Vision I*. Skriptum zur Vorlesung, Sommersemester 2002, Universität Ulm, Fakultät für Informatik. 2002
- [Neu02b] NEUMANN, Heiko. *Computer Vision II*. Skriptum zur Vorlesung, Wintersemester 2002/2003, Universität Ulm, Fakultät für Informatik. 2002
- [NF90] NOVIKOV, Sergej P. ; FOMENKO, Anatolij T.: *Basic Elements of Differential Geometry and Topology*. 1. Auflage. Dordrecht : Kluwer Academic Publishers, 1990. – 490 S. – ISBN 0-7923-1009-8
- [NM01] NEUMANN, Heiko ; MINGOLLA, Ennio: Computational Neural Models of Spatial Integration in Perceptual Grouping. In: SHIPLEY, Thomas F. (Hrsg.) ; KELLMAN, Philip J. (Hrsg.): *From Fragments to Objects*. 1. Auflage. Amsterdam : Elsevier, 2001 (Advances in Psychology). – ISBN 0-444-50506-7, Kapitel 12, S. 353-400
- [NM02] NICOLESCU, Mircea ; MEDIONI, Gérard: Perceptual Grouping from Motion Cues Using Tensor Voting in 4-D. In: HEYDEN, Anders (Hrsg.) ; SPARR, Gunnar (Hrsg.) ; NIELSEN, Mads (Hrsg.) ; JOHANSEN, Peter (Hrsg.): *Computer Vision - ECCV 2002, Vol. III*. Berlin, Heidelberg : Springer-Verlag, Mai 2002 (Lecture Notes in Computer Science Vol. 2352). – ISBN 3-540-43746-0, S. 423-437
- [Ots79] OTSU, Nobuyuki: A Threshold Selection Method from Gray-Level Histograms. In: *IEEE Transactions on Systems, Man, and Cybernetics* 9 (1979), Januar, Nr. 1, S. 62-66
- [RK82] ROSENFELD, Azriel ; KAK, Avinash C.: *Digital Picture Processing*. Bd. 2. 2. Auflage. Orlando : Academic Press, 1982. – 349 S. – ISBN 0-12-597301-2
- [RN95] RUSSELL, Stuart J. ; NORVIG, Peter: *Artificial Intelligence: A Modern Approach*. 1. Auflage. Upper Saddle River, NJ : Prentice Hall, Februar 1995. – 932 S. – ISBN 0-13-360124-2
- [Rot01] ROTTENSTEINER, Franz: *Semi-automatic extraction of buildings based on hybrid adjustment using 3D surface models and management of building data in a TIS*, Technische Universität Wien, Diss., Februar 2001
- [SB97] SMITH, S. M. ; BRADY, J. M.: SUSAN - A New Approach to Low Level Image Processing. In: *International Journal of Computer Vision* 23 (1997), Mai, Nr. 1, S. 45-78
- [SB01] SPREEUWERS, Luuk ; BREEUWER, Marcel: Automatic Detection of the Myocardial Boundaries of the Right and Left Ventricle in MR Cardio Perfusion Scans. In: *Proceedings of the SPIE Conference on Medical Imaging*. San Diego, USA, Februar 2001, S. 1207-1217

Literaturverzeichnis

- [SG02] STEGMANN, Mikkel B. ; GOMEZ, David D.: *A Brief Introduction to Statistical Shape Analysis*. Informatics and Mathematical Modelling, Technical University of Denmark, DTU. März 2002. – 15 S.
- [Soj02] SOJKA, Eduard: A New Algorithm for Direct Corner Detection in Digital Images / Technical University of Ostrava. Ostrava-Poruba, Czech Republic, September 2002. – Forschungsbericht
- [Soj03] SOJKA, Eduard. *A New Algorithm for Detecting Corners in Digital Images*. URL: <http://www.cs.vsb.cz/sojka/cordet/presentation.html>. 2003
- [Sör96] SÖRCEL, Wolfgang: *Model-based Segmentation of 4D MR Images of the Heart*, Universität Erlangen-Nürnberg, Diplomarbeit, 1996. – URL: <http://www.nt.e-technik.uni-erlangen.de/~wsoergel/publications/dipl.ps.gz>
- [Ste01] STEUP, Dörte: *Untersuchungen zur automatischen Erfassung der Straßengeometrie für die Aktualisierung digitaler Straßenkarten*, Technische Universität Dresden und Universität Stuttgart, Diplomarbeit, September 2001. – <http://www.uni-stuttgart.de/iagb/publikation/darbeit/doerte/all/node40.html>
- [Tan00] TANG, Chi-Keung: *Tensor Voting in Computer Vision, Visualization, and Higher Dimensional Inferences*, University of Southern California, Diss., Januar 2000
- [TM98] TANG, Chi-Keung ; MEDIONI, Gérard: Integrated Surface, Curve and Junction Inference from Sparse 3-D Data Sets. In: *Proceedings of International Conference on Computer Vision*. Bombay, Indien, Januar 1998, S. 818–824
- [TM02] TANG, Chi-Keung ; MEDIONI, Gérard: Curvature-Augmented Tensor Voting for Shape Inference from Noisy 3D Data. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002), Juni, Nr. 6, S. 858–864
- [TMV99] THEWS, Gerhard ; MUTSCHLER, Ernst ; VAUPEL, Peter: *Anatomie, Physiologie, Pathophysiologie des Menschen*. 5. Auflage. Stuttgart : Wissenschaftliche Verlagsgesellschaft, 1999. – 868 S. – ISBN 3–8047–1616–4
- [Wer23] WERTHEIMER, Max: Untersuchungen zur Lehre von der Gestalt II. In: *Psychologische Forschung*, 4, 1923. – <http://psy.ed.asu.edu/~classics/Wertheimer/Forms/forms.htm>, S. 301–350
- [WS00] WANG, Yongmei ; STAIB, Lawrence H.: Boundary Finding with Prior Shape and Smoothness Models. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (2000), Nr. 7, S. 738–743
- [XP98] XU, Chenyang ; PRINCE, Jerry L.: Snakes, Shapes, and Gradient Vector Flow. In: *IEEE Transactions on Image Processing* 7 (1998), März, Nr. 3, S. 359–369
- [XP99] XU, Chenyang ; PRINCE, Jerry L. *Active Contours and Gradient Vector Flow - Software*. URL: <http://iacl.ece.jhu.edu/projects/gvf/>. 1999

A

Ableitung, 6, 23, 24
Abschwächungsfunktion, 18–20, 59, 64
Achternachbarschaft, 51
Active Appearance Model, 4
Active Contour Model, 55
Active Shape Model, 3, 69, 71
Anatomie, 45
Anti-Aliasing, 37
Aorta, 46
Artefakte, 32, 33, 36, 38, 40, 41, 43, 51
ASM, *siehe* Active Shape Model
Assoziationsfeld, 15
Atmungsbewegungen, 46
Atrium, 46
Ausreißer, 61, 63, 64

B

Ballkomponente, 11
Basis-Tensor, 29
Beaudet-Operator, 25
Berechnungsaufwand, *siehe* Laufzeit
Bewegungsartefakte, 46, 59
Bildrestauration, 4
Bildstörungen, 12
Bottom-Up, 2, 3, 69, 76
Boxfilter, 26
Brustbein, 46

C

Canny-Operator, 24, 39, 44
CMap, 15
Constraint Satisfaction Problem, 3

D

Diastole, 1, 46
Differentialgeometrie, 10
Diskretisierungseffekte, 37
Durchblutung, 1, 48
dyadisches Produkt, 15

E

Ecke, 4
Eckendetektion, 24
 Beaudet, 24
 Deriche und Giraudon, 25
 Dreschler und Nagel, 25
 Harris, 26
 Noble, 26
Einflussfaktoren, 32
Eingabemerkmale, 7
 Flächenelement, *siehe* Flächenelement
 Kurvenelement, *siehe* Kurvenelement
EKG, *siehe* Elektrokardiogramm
Elastizität, 55, 68
Elektrokardiogramm, 46
elliptisch, 25
Endokontur, 1, 53
Endpunkt, 4
Energie, 17
Epikontur, 1, 53

F

Faltung, 12
Fan-In, 20
Fan-Out, 20
Flächenelement, 7
Form, 71

Index

Funktional, 3, 55

G

Gabor-Filter, 28
Gauß-Funktion, 17, 28
Gauß'sche Krümmung, 24
Gauß'sches Rauschen, 35
Gestaltpsychologie, 16
Gradient Vector Flow, 55
Grauwertgradient, 6, 21
Ground truth, *siehe* Grundwahrheit
Grundwahrheit, 31, 32
Gruppierungsfähigkeit, 20

H

Hauptkomponentenanalyse, 72
Hauptkrümmung, 10, 24
Hauptschlagader, *siehe* Aorta
Herz, 45
Herzkammer, *siehe* Ventrikel
Hesse-Matrix, 24
Hohlvene, *siehe* Vena cava
Hough-Transformation, 2
hyperbolisch, 25
hysteresis thresholding, 24

I

Illusionäre Kontur, 21
Intensitätsmaske, 52
Interpolation
 bilineare, 79
isotrop, 7, 19

J

JMap, 15

K

Kanisza Figuren, 21
Kante, 5
 rampenförmige, 40
Stufen-, *siehe* Stufenkante
Kantendetektion, 24
 Canny, 24, 39
 Laplace von Gauß, 24
 Prewitt, 24
 Robert, 24

Sobel, 24

Konsistentes Labeling, 3

Kraft

 äußere, 55

 Druck, 55

 innere, 55

Kreuzungspunkt, 4

Krümmung, 10, 18

 Gauß'sche, *siehe* Gauß'sche Krümmung

 Haupt-, *siehe* Hauptkrümmung

 Total-, *siehe* Totalkrümmung

Kurvenelement, 7

L

Landmarke, 71

Laufzeit, 22, 48, 60

Linie, 5

Lungenschlagader, *siehe* Truncus pulmonalis

Lungenvenen, *siehe* Venae pulmonales

M

Magnetresonanztomographie, 1

Mahalanobis-Distanz, 75

Mediastinum, 46

Merkmal

 Eingabe-, *siehe* Eingabemerkmale

 Struktur-, *siehe* Strukturmerkmale

Merkmalsextraktion, 21

 einfache, 24

 polymorphe, 26

Mittelfellraum, *siehe* Mediastinum

Modellform-Parameter, 73

Modellwissen, 2, 48, 71

Myokard, 46

N

Non-Maximum Suppression, 21, 24

Normalverteilung, 20

Notation, 6

O

Ösophagus, 46

P

parabolisch, 25

Pathophysiologie, 45

- PCA, *siehe* Principal Component Analysis
 Phase, 46
 Physiologie, 45
 Plessey-Detektor, *siehe* Eckendetektion, Harris
 Polarität, 65
 Polaritätssalienz, 65
 Positionsfehler
 quadratischer, *siehe* Quadratischer Positionsfehler
 Principal Component Analysis, 72
 Prokrustes-Transformation, 72
- Q**
 Quadratischer Positionsfehler, 32
- R**
 Rauschen, 35
 Referenzform, 72
 Region, 5
 Regularisierung, 3
 Rigidität, 55, 68
- S**
 Salienz, 10
 Salienzkarte, 9, 15
 Scheibenkomponente, 11
 Schwellwert, 22, 24
 Schwellwertverfahren, 22, 24
 Shot Noise, 35, 51
 Skalenfaktor, 20, 33, 69, 70
 Skalenraum, 20
 Snakes, 3, 55
 Sobel-Operator, 23, 40
 Stabkomponente, 11
 Sternum, 46
 Stimme, 14, 60
 Strukturtensor, 26, 39
 Stufenkante, 40
 Systole, 1, 46
- T**
 Tensor, 7, 10–11
 Komponentendarstellung, *siehe* Tensorkomponente
 Tensorkarte, 7
 Tensorkodierung, 12
 Tensorkomponente, 11
- Ballkomponente, 11
 Scheibenkomponente, 11
 Stabkomponente, 11
 Tensorvotierung, 12
 dichte, 14
 spärliche, 14
 Tensorzerlegung, 14
 Top-Down, 3, 71
 Totalkrümmung, 16
 Truncus pulmonalis, 46
- U**
 Umrisslinie, 1, 5, 50, 53, 65
- V**
 Varianzmaske, 50
 Vena cava, 46
 Venae pulmonales, 46
 Ventrikel, 46
 Verbindungslinie, 13, 15, 16, 18
 Vierernachbarschaft, 51
 Viskosität, 55, 68
 Volumenelement, 70
 Vorhof, *siehe* Atrium
 Votierfeld, 15
 Ball-, 19
 dreidimensionales, 20
 für Krümmung, 63, 64
 grundlegendes, 16
 Anforderungen, 16
 Gesamtformel, 18–19
 Orientierung, 16
 Stärke, 17–18
 Voxel, 70
- W**
 Wahrnehmungsgruppierung, 2, 15
- Z**
 Zeitkomplexität, 22
 zero-crossings, 24
 Zwerchfell, 46

Index

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt,
dass ich die vorliegende Diplomarbeit selbständig
und ohne unzulässige fremde Hilfe angefertigt habe.

Alle verwendeten Quellen und Hilfsmittel sind angegeben.

Ulm, den __. November 2003

Axel Gerstmair